

COMPUTER VISION BASED EARLY FIRE DETECTION USING MACHINE LEARNING

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering Degree in Computer Science and Engineering

by

KONDA SRAVAN KUMAR (38110258)
KUNAPARAJU BALARAM SAI KUMAR (38110279)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119.

MARCH 2022



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade “A” by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
Chennai - 600119
www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KONDA SRAVAN KUMAR (Reg. No. 38110258), KUNAPARAJU BALARAM SAI KUMAR (Reg. No. 38110279)** who carried out the project entitled “**COMPUTER VISION BASED EARLY FIRE DETECTION USING MACHINE LEARNING TECHNIQUES**” under our supervision from Nov 2019 to April 2020.

Internal Guide

Dr. L. SUJHELEN, M.E., Ph.D.,

Head of the Department

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

I, **KONDA SRAVAN KUMAR (Reg. No. 38110258), KUNAPARAJU BALARAM SAI KUMAR (Reg. No. 38110279)** hereby declare that the Project Report entitled **“COMPUTER VISION BASED EARLY FIRE DETECTION”** done by me under the guidance of **Dr. L. SUJI HELEN, M.E., Ph.D.**, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey our thanks to **Dr. T. SASIKALA M.E., Ph.D., Dean, School of Computing, Dr. S. Vigneshwari M.E., Ph.D. and Dr. L. Lakshmanan M.E., Ph.D., Heads of the Department**, Department of Computer Science and Engineering for providing us the necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. L. SUJHELEN, M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of **COMPUTER SCIENCE AND ENGINEERING** who were helpful in many ways for the completion of the project.

ABSTARCT

The project aimed to detect fire by using the image processing technology that will alert people by early detection of fire. As there are many automatic fire alarm systems already existed like the sensor method, that has some limitations and designed to sense fire with the smoke, limited areas. To reduce limitations and to optimize with new technology, the project is proposed. The project is implemented 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 entire code is written in pure 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 gives a better understanding of object detection with the computer and the use of these technologies in different forms and uses.

NUMBER	TITLE	PAGE NUMBER
	ABSTRACT	v
	LIST OF FIGURES	viii
	LIST OF TABLES	ix
1	INTRODUCTION	1
	1.1 OVERVIEW	1
	1.2 PURPOSE OF MACHINE LEARNING	2
	1.3 PROBLEM STATEMENT	2
	1.4 OBJECTIVES	3
2	LITERATURE SURVEY	4
3	SYSTEM ANALYSIS	7
	3.1 DRAWBACKS OF EXISTING SYSTEM	7
	3.2 PROPOSED SYSTEM	7
	3.3 SYSTEM REQUIREMENTS	8
	3.3.1 HARDWARE REQUIREMENTS	8
	3.3.2 SOFTWARE REQUIREMENTS	8
4	SYSTEM DESIGN	9
	4.1 PROPOSED WORK	9
	4.2 ARCHITECTURE DIAGRAM	11
	4.3 DATA SET	13
	4.4 ALGORITHMS AND METHODS	14
	4.4.1 DECISION TREE	14
	4.4.2 CNN	17

5	SYSTEM IMPLEMENTATION	19
	5.1 MODULES	19
	5.1.1 PRE-PROCESSING OF AN IMAGE	19
	5.1.2 SEGMENTATION OF AN IMAGE	20
	5.1.3 FEATURE EXTRACTION OF AN IMAGE	21
	5.1.4 MACHINE LEARNING CLASSIFIER	22
	5.2 RESULTS AND DISCUSSION	23
	5.2.1 RESULTS	23
	5.3 EXPERIMENTAL RESULTS	29
6	CONCLUSION	31
	6.1 CONCLUSION	31
	6.2 FUTURE ENHANCEMENT	32
	REFERENCES	33
	APPENDIX	35
	A. SOURCE CODE	35

Chapter 1

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 Purpose of machine learning

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly

programmed to do so. Machine learning algorithms use historical data as input to predict new output values. an algorithm would be trained with pictures of dogs and other things, all labeled by humans, and the machine would learn ways to identify pictures of dogs on its own. Supervised machine learning is the most common type used today.

1.3 Problem statement

- The main motivation for fire prediction is to provide proper resource allocation and to help in best possible way to firefighters of Fire Management team.
- The main factors of fire are Meteorological conditions. the climatic information is gotten from nearby sensors which are fused in the closest meteorological stations.
- Land with a possible high fire risk has many indicators that can be used to measure the forecast by closely evaluating the indications.
- Every year, fire destroys millions of hectares of land. These fires have burned vast areas and generate more carbon monoxide than total vehicle traffic.
- Monitoring potential danger areas and early warning of fire can greatly reduce response time, as well as the potential for damage and firefighting costs.

1.4 Objective

The four ways your fire alarm system works to protect your property and its occupants from the dangers of fire are by detecting fire, alerting occupants, managing risk, and notifying the fire department.

Chapter 2

LITERATURE REVIEW:

2.1 A Video-Based Fire Detection Using Deep Learning Models

Abstract:

Fire is an abnormal event which can cause significant damage to lives and property. In this paper, we propose a deep learning-based fire detection method using a video sequence, which imitates the human fire detection process. The proposed method uses Faster Region-based Convolutional Neural Network (R-CNN) to detect the suspected regions of fire (SRoFs) and of non-fire based on their spatial features. Then, the summarized features within the bounding boxes in successive frames are accumulated by Long Short-Term Memory (LSTM) to classify whether there is a fire or not in a short-term period. The decisions for successive short-term periods are then combined in the majority voting for the final decision in a long-term period. In addition, the areas of both flame and smoke are calculated and their temporal changes are reported to interpret the dynamic fire behavior with the final fire decision. Experiments show that the proposed long-term video-based method can successfully improve the fire detection accuracy compared with the still image-based or short-term video-based method by reducing both the false detection's and the misdetections.

Introduction

Fire is an abnormal event which can quickly cause significant injury and property damage. According to the National Fire Protection Association (NAPA), the United States fire department responded to an estimated 1,319,500 fires during 2017, which resulted in 3,400 civilian fire fatalities, 14,670 civilian fire injuries, and an estimated \$23 billion in direct property loss. In order to reduce such disasters, fire detection without a false alarm at an early stage is crucial. Accordingly, various automatic fire detection technologies are being developed, and are widely used in real life. In general, two broad categories of technologies can be identified: traditional fire alarm and fire detection by computer vision. Traditional fire alarm technology is based on smoke or heat sensors that require proximity for activation. These sensors need human involvement to confirm a fire in case of alarm. Furthermore, such systems require various equipment to provide information on the size, location, and burning degree of the fire. To overcome these limitations, researchers have been investigating computer vision-based methods combined with various types of supplementary sensors. This category of technologies gives larger surveillance coverage and offers

the advantage of less human intervention with a faster response, as a fire can be confirmed without requiring a visit to the fire location, and provides detailed fire information such as location, size, and degree. Despite these advantages, however, some issues remain concerning the system complexity, and false detection according to diverse reasons. Therefore, researchers have invested significant effort to address these issues in terms of computer vision technology. Early research on computer vision-based fire detection was focused on the color of a fire within the framework of a rule-based system, which is often sensitive to environmental conditions such as illumination and weather. So, further studies added supplementary features to the color of a fire, including area, surface, boundary, and motion of the suspected region, with other types of decision-making algorithms, such as Bayes classifier and multi-expert systems, in order to make a robust decision. Nevertheless, almost all the research tries to detect the flame and smoke in a single frame of Closed-Circuit Television (CCTV) or a limited number of frames in a short period. In general, it is not an easy task to explore the static and dynamic characteristics of diverse flame and smoke to be exploited in a vision system, as it requires a large amount of domain knowledge. In the deep learning approach however, these exploration and exploitation processes can be replaced by the training of an appropriate neural network with a sufficient amount of data in order to avoid overfitting. This approach, therefore, becomes convenient once a dataset with many flame and smoke images or video clips has been built. In this paper, we propose a deep-learning-based fire detection method, which imitates the human detection process and which we call the detection and temporal accumulations (DTA) for the fire decision. Usually, a human can detect a suspected fire object in a scene, continuously monitor it, and accumulate the temporal behaviors to finally decide whether it is a fire or not. We assume that this DTA process can greatly reduce erroneous fire detection. In the proposed method, the suspected region of fire (SRoF) is detected with its spatial features against non-fire objects by the Faster Region-based Convolutional Neural Network (R-CNN). Then, the features summarized from the object detection model in successive frames are accumulated by Long Short-Term Memory (LSTM) to classify whether there is a fire or not in a short-term period, which can be treated as a person's glance for fire detection. The decisions for successive short-term periods are then combined in the majority voting for the final decision in a long-term period. In addition, the areas of SRoFs, including both flame and smoke, are calculated and their temporal changes are

reported to interpret the dynamic fire behavior with the final fire decision.

Experiments show that the proposed long-term video-based method can successfully improve the fire detection accuracy compared with the still image-based or short-term video-based method by reducing both the false detections and the misdetections. The method discriminates fires from fire-like video sequences especially well. For example, chimney smoke, sunset, and clouds often induce errors in conventional computer vision-based fire detection. Also, the method reflects the temporal behavior of real fire situations well by providing area information of SROFs. Therefore, our key contributions can be summarized as follows:

- (1) We propose a deep learning-based fire detection method that avoids the time-consuming efforts to explore hand-crafted features. Because it automatically generates a set of useful features after training, it is sufficient to construct the proper deep learning model and to gather a sufficient amount of training data. Therefore, we have constructed a large fire dataset which contains diverse still images and video clips, including the data from well-known public datasets. Not only is the dataset used for the training and testing of our experiment, but it also could be an asset for future computer vision-based fire detection research.
- (2) Our deep learning-based method emulates a human process of fire detection called DTA, in that SROFs are detected in one scene and the temporal behaviors are continuously monitored and accumulated to finally decide whether it is a fire or not. In the method, Faster R-CNN is used to detect SROFs against non-fire objects with their spatial features, and LSTM temporally accumulates the summarized spatial features by using the weighted Global Average Pooling (GAP), where the weight is given by the confidence score of a bounding box. The initial decision is made in a short period, and the final decision is made by the majority voting of the series of decisions in a long period.
- (3) The proposed method has been experimentally proven to provide excellent fire detection accuracy by reducing the false detections and misdetections. Also, it successfully interprets the temporal SROF behavior, which may reduce false dispatch of firemen

Related Work

Computer Vision-Based Fire Detection

In conventional fire detection, much research has continuously focused on finding out the salient features of fire images. Chen analyzed the changes of fire using an RGB and HSI color model based on the difference between consecutive frames and proposed a rule-based approach for fire decision. Celik and Demirel proposed a generic rule-based flame pixel classification using the YCbCr color model to separate chrominance components from luminance ones. In addition, Wang extracted the candidate fire area in an image using an HSI color model and calculated the dispersion of the flame color to determine the fire area. However, color-based fire detection methods are generally vulnerable to a variety of environmental factors such as lighting and shadow. Borges and Izquierdo adopted the Bayes classifier to detect fires based on additional features such as the area, surface, and boundary of the fire area to color. Mueller proposed the neural network-based fire detection method using optical flow for the fire area. In the method, two optical flow models are combined to distinguish between fire and dynamically moving objects. In addition, Foggia proposed a multi-expert system which combines the analysis results of a fire's color, shape, and motion characteristics. Although insufficient, the supplementary features to color, including texture, shape, and optical flow, can reduce the false detections. Nevertheless, these approaches require domain knowledge of fires in captured images essential to explore hand-crafted features and cannot reflect the information spatially and temporally involved in fire environments well. In addition, almost all methods using the conventional approach only use a still image or consecutive pairs of frames to detect fire. Therefore, they only consider the short-term dynamic behavior of fire, whereas a fire has a longer-term dynamic behavior.

Deep Learning-Based Approach

Recently, deep learning has been successfully applied to diverse areas such as object detection/classification in images, speech recognition, and natural language processing. Researchers have conducted various studies on fire detection based on deep learning to improve performance. The deep learning approach has several differences from the conventional computer vision-based fire detection. The first is that the features are not explored by an expert, but rather are automatically captured in the network after training with a large amount of diverse training data. Therefore, the effort to find the proper handcrafted features is shifted to designing a

proper network and preparing the training data. Another difference is that the detector/classifier can be obtained by training simultaneously with the features in the same neural network. Therefore, the appropriate network structure becomes more important with an efficient training algorithm. Sebastien proposed a fire detection network based on CNN where the features are simultaneously learned with a Multilayer Perceptron (MLP)-type neural net classifier by training. Zhang et al. also proposed a CNN-based fire detection method which is operated in a cascaded fashion. In their method, the full image is first tested by the global image-level classifier, and if a fire is detected, then a fine-grained patch classifier is used for precisely localizing the fire patches. Muhammad et al. proposed a fire surveillance system based on a fine-tuned CNN fire detector. This architecture is an efficient CNN architecture for fire detection, localization, and semantic understanding of the scene of the fire inspired by the Squeeze Net. In the deep layer of CNN, a unit has a wide receptive field so that its activation can be treated as a feature that contains a large area of context information. This is another advantage of the learned features with CNN for fire detection. Even though CNN showed overwhelmingly superior classification performance against traditional computer vision methods, locating objects has been another problem. In the proposed method, we adopt the object detection model to localize the SRoFs and non-fire objects, which includes the flame, smoke for the SRoFs, and other objects irrelevant to the fire for the non-fire objects. The objects irrelevant to the fire increase false alarms due to variations in shadows and brightness, and will often detect objects such as red clothes, red vehicles, or sunset. We detect the fire objects by using the Faster R-CNN model, even though it does not have to be confined to the object detection model. The deep object detector, either single- or multi-stage, is usually composed of CNN-type feature extractors, followed by a localizer with a classifier. Therefore, our object detection model includes the feature extractor with a relatively wider area of receptive field than the detected SRoF area and can gather more context information.

Although the CNN-based approaches provide excellent performance, it is hard to capture the dynamic behavior of fire, which can be obtained by recursive-type neural networks (RNN). LSTM proposed by Hochreiter and Schmidhuber is an RNN model that solves the vanishing gradient problem of RNN. LSTM can accumulate the temporal features for decision making through the memory cells which preserve the internal states and the recurrent behavior. However, the number of recursions is

usually limited, which makes it difficult to capture the long-term dynamic behavior necessary to make a decision. Therefore, special care must be taken to consider the decision based on long-term behavior with LSTM. Recently, Hu et al. used LSTM for fire detection, where the CNN features are extracted from optical flows of consecutive frames, and temporally accumulated in an LSTM network. The final decision is made based on the fusion of successive temporal features. Their approach, however, computes the optical flow to prepare the input of CNN rather than directly using RGB frames.

Proposed Method

Network Architecture

Traditional computer vision-based fire detection methods have widely used the static characteristics or the short-term temporal behaviors such as colors and motions of flame and smoke. However, as fires show variable temporal appearance, the detection accuracy of such methods that depend on the static and short-term temporal behaviors is limited. We propose a deep learning-based fire detection method, which imitates the human process, called DTA for fire decision. We assume that this DTA process can greatly reduce erroneous fire decisions. The proposed network architecture is divided into three sections. In the first section, we detect the SRoFs or non-fire objects in the video frames using a deep object detection model, Faster R-CNN, which consists of CNN feature extractors and a bounding box localizer with a classifier. Here, the bounding boxes locate three different classes: flame, smoke, and non-fire. Usually, flame and smoke cannot be well-separated in a fire so that the smoke-only object in a bounding box is classified into smoke. Also, the whole fire region is treated as one bounding box. A non-fire object implies a still image that has no objects related with a fire or a class of objects that are difficult to differentiate from a fire, such as a chimney evening glow, smoke, and cloud. The non-fire objects have their own bounding boxes. Then, the bounding boxes, including SRoFs and non-fire objects, are projected on the learned feature maps in the last layer of CNN of Faster R-CNN in order to extract the corresponding spatial features. In the second section, the summarized and concatenated CNN features are temporally accumulated to capture the dynamic behaviors of fire, and the short-term fire decision is made in the two-stage LSTM network. Here, we do not differentiate flame from smoke in the section, so that the LSTM consecutively accumulates both flame and smoke features

to decide between fire or non-fire. Then, in the third section, the short-term decisions are combined in the last majority voting stage for long-term fire decision. The last block also integrates the information to interpret the dynamic fire behaviors in order to determine whether the area of SRoFs, including flame and smoke detected by bounding boxes at Faster R-CNN stages, is increasing or not, for the long-term period. the proposed network architecture.

Fire Object Detection Based on Faster Region-Based Convolutional Neural Network (R-CNN)

Faster R-CNN is a CNN-based object detection method that combines both the Fast R-CNN and the Region Proposal Network (RPN) to share a convolutional network after excluding the fully connected layer. So Faster R-CNN shares a similar structure for object detection with Fast R-CNN, except that Faster R-CNN includes RPN to generate region proposals for objects. Based on the proposals, Faster R-CNN extracts the spatial features through the ROI pooling operation, and then calculates the object positions with class scores by fully connected layers. Usually, Faster R-CNN provides a higher mean Average Precision (mAP) than the single-stage object detection models such as SSD (Single Shot Multibox Detector) and YOLO (You Only Look Once). Reportedly, Faster R-CNN showed mAP as high as 34.9% for the MS-COCO dataset, when the shared CNN feature extractor is equipped with ResNet 101, which we have adopted in our work. In our method, Faster R-CNN provides the bounding boxes of flame, smoke, and non-fire regions in an image, represents the sample images of flame, smoke, and non-fire objects. The non-fire objects resemble real fire objects, such as chimney smoke, sunset, and cloud. In addition, the image containing objects that are not related to a fire is itself treated as a non-fire object.

When the Faster R-CNN detects the classes of flame, smoke, and non-fire objects, false detections may arise due to the possible presence of various types of non-fire objects in a single frame that is similar to a fire. The non-fire objects that resemble fires include sunset, chimney smoke, cloud, etc.

As aforementioned, however, applying this deep object detection model to find SRoFs and non-fire objects offer an advantage. Because the consecutive convolution enlarges the effective area of the operation, a bounding box resulting from the deep object detection model can include the larger area of a receptive field

than it encloses. So, the more context information around objects can be captured in the boxes. This implies that SRoF detection becomes robust because it better reflects the context information around the bounding box.

In our proposed method, a weighted Global Average Pooling (GAP) scheme is adapted to extract the spatial features. After insignificant bounding boxes are filtered out by thresholding their own confidence score, the significant SRoFs and non-fire objects are selected. The image which does not contain any specific small bounding box is treated as a non-fire object whose bounding box covers the whole image with confidence score 1. Note that the significant SRoF or the non-fire object has its own confidence score which can be used to take the weighted GAP. Figure 5 shows the process to extract the spatial features from the last layer of CNN of the Faster R-CNN object detector with d feature maps, where $d = 1024$. From each feature map f_i ,

The vector $\mathbf{v} = (v_1, v_2, \dots, v_d)$ represents the aggregated spatial feature for SRoFs or non-fire objects detected by Faster R-CNN in an image or a frame of a video. In general, the prominent features among d can be found projecting the bounding box of SRoFs or non-fire objects on the feature map similar to the class activation map. Because they are merely spatial features which do not contain temporal information, the feature selection in our proposed method is transferred to the following LSTM stage of the temporal aggregation in a short-term.

Long Short-Term Memory (LSTM) Network for Fire Features in a Short-Term

In general, it is not appropriate to detect and judge the fire without considering the temporal behavior. In the proposed method we aggregate the changes in the extracted spatial features using LSTM in a short period $TLSTM$, and try to determine whether it is a fire or a non-fire object. Here, we do not differentiate between flame and smoke. Because the SRoFs or non-fire objects in consecutive frames have been determined by Faster R-CNN, we merely temporally accumulate the spatial features within the corresponding bounding boxes in LSTM and determine whether the consecutive box is fire or not for the short-term. This important step in DTA in the proposed method is similar to a person's quick glance to detect a fire. We assume that the fire decision based on a glance depends on the short-term dynamic characteristics of the fire.

In the proposed method, the LSTM network consists of two stages in which the number of memory cells in LSTM is determined experimentally. The short temporal features

pooled through the LSTM network are used to make a short-term fire decision by two soft-max units, one for fire or the other for non-fire. Figure 7 shows the part of LSTM used to accumulate and decide a fire in a short time period.

Faster R-CNN showed mAP as high as 34.9% for the MS-COCO dataset, when the shared CNN feature extractor is equipped with ResNet 101, which we have adopted in our work.

In our method, Faster R-CNN provides the bounding boxes of flame, smoke, and non-fire regions in an image, as shown in Figure 3. Figure 4 represents the sample images of flame, smoke, and non-fire objects. The non-fire objects resemble real fire objects, such as chimney smoke, sunset, and cloud. In addition, the image containing objects that are not related to a fire is itself treated as a non-fire object.

Experiments and Results

Our method does not use end-to-end training because the weighted GAP within bounding boxes and the majority voting processes are included. Because both of them are non-differentiable operations, Faster R-CNN and LSTM stages should be separately trained in the proposed method.

Training Faster R-CNN and Its Accuracy

Faster R-CNN requires still images for training data so we collected them from several data sources. Some fire and smoke images were taken from Youtube video clips. Also, the same data as the previous works were added, which include still images and frames taken from video clips. In addition, the Flickr-fire dataset was included in our dataset. Finally, we constructed a dataset of 73,887 still images, consisting of 22,729 flame, 23,914 smoke, and 27,244 non-fire images. The images are divided into 75% for training, 15% for validation, and 10% for test data. For training, the data are augmented by a horizontal flip. training parameters for Faster R-CNN.

The performance of the Faster R-CNN is measured by map and is shown in The sample results of the flame and smoke detection. There are several false positive detections for clouds, chimney smoke, lighting lamp, steam, etc., which are almost undetectable without considering the temporal characteristics.

Training LSTM and Its Performance

The LSTM in the proposed method is trained with video clips. We collected 1,309 video clips from Youtube, comprising 672 clips of fire, and 637 of non-fire. As in the Faster R-CNN, the video clips of non-fire objects can include hard negative examples like clouds, chimney smoke, lighting lamp, and steam, or simply objects that are irrelevant to fire. shows samples of flame, smoke, and non-fire objects from the LSTM training video dataset.

Here, we do not discriminate between flame and smoke as mentioned before. The video clips are divided into 60 consecutive frames with 15 frames of overlap that last for about 2 s if 30 frames per second are assumed. This implies the LSTM network captures short-term dynamic behaviors of a fire or non-fire and decides whether it is a fire or not for every 1.5 s. Here we assume a person's quick glance for fire decision happens every 1.5 s. The time duration can be adjusted according to the situation when our method is implemented.

For the LSTM training, we prepared 8,527 positive and 7,547 negative examples of 60 frame video clips from Youtube videos. From the examples, 75% of the data were selected for training, 15% for validation, and 10% for testing. From each video clip, we obtained bounding boxes and their corresponding 1024-dimensional feature for every consecutive frame, which gave a set of sequential inputs to LSTM. the parameters of LSTM training and the performance of the test data shown in. according to the number of memory cells in LSTM. To compare the performance with another method, we evaluated the results using the dataset in reference. Information of the public dataset which consists of 31 video clips under different conditions, and the still shots taken from several samples.

Majority Voting and Interpretation of Fire Behavior

The LSTM short-term fire decisions during T_{vot} are involved in the majority voting for the final fire decision. Because only short video clips are included in the dataset for comparison, we take $T_{\text{vot}} = 10$ s for majority voting. Even with such a short-term ensemble, the accuracy increases by up to 97.92%.

Also, we collected an additional 38 video clips from the internet, including Youtube, which have a relatively long playing-time. the categorized fire/non-fire video clips with their time-varying behaviors. samples of the video dataset. We performed the majority voting for the final fire decision and evaluated the accuracy according to the time period of T_{vot} and the results are summarized. Note that the longer time period provides better accuracy in general because more LSTM decisions are combined in the majority voting to make a robust ensemble. However, the dispatch of firemen should be done as early as possible so that T_{vot} can be adjusted by the trade-off between the accuracy and the critical time for dispatch.

We have compared the performances of our scheme in terms of three metrics including false positive, false negative, and accuracy. While the method of Khan Muhammad et al. performs the best in terms of false positive and false negative, ours with the delayed decision of the majority voting in 10s outperforms in accuracy. Note that our proposed method can produce this better by introducing the delayed decision in DTA.

Conclusions

We have proposed a deep learning-based fire detection method, called DTA, which imitates the human detection process. We assumed that the DTA process can greatly reduce erroneous fire detection. The proposed method uses the Faster R-CNN fire detection model to detect the SRoF based on its spatial features. Then, the features summarized from the SRoFs and non-fire regions in successive frames are accumulated by LSTM to classify whether there is a fire or not in a short-term period. The successive short-term decisions are then combined in the majority voting for the final decision in a long-term period. In addition, the areas of both flame and fire are calculated and their temporal changes are reported to interpret the dynamic behavior of the fire with the final fire decision.

The proposed method has been experimentally proven to provide excellent fire detection accuracy, by reducing the false detections and misdetections, and to successfully interpret the temporal behavior of flame and smoke, which possibly reduces the false dispatch of firemen. In addition, we have constructed a large fire dataset which contains diverse still images and video clips that enhance the data from well-known

public datasets. Not only is the dataset used for the training and testing of our experiment, but it also could be an asset for future fire research.

2.2 Computer Vision-based Early Fire Detection Using Enhanced Chromatic Segmentation and Optical Flow Analysis Technique

Abstract:

Recent advances in video processing technologies have led to a wave of research on computer vision-based fire detection systems. This paper presents a multi-level framework for fire detection that analyses patterns in chromatic information, shape transmutation, and optical flow estimation of fire. First, the decision function of fire pixels based on chromatic information uses majority voting among state-of-the-art fire color detection rules to extract the regions of interest. The extracted pixels are then verified for authenticity by examining the dynamics of shape. Finally, a measure of turbulence is assessed by an enhanced optical flow analysis algorithm to confirm the presence of fire. To evaluate the performance of the proposed model, we utilize videos from the Mivia and Zenodo datasets, which have a diverse set of scenarios including indoor, outdoor, and forest fires, along with videos containing no fire. The proposed model exhibits an average accuracy of 97.2% for our tested dataset. In addition, the experimental results demonstrate that the proposed model significantly reduces the rate of false alarms compared to the other existing models.

Introduction

Fire is a common disaster that claims thousands of lives and does insurmountable damage to property every year. Early fire detection has, therefore, become increasingly important to protect against this hazard. Even though sensor-based fire detection systems relying on heat or smoke signals are ubiquitous, these systems have multiple drawbacks. First, sensor-based detectors have to be densely distributed because they have limited coverage in terms of area. This results in raised expenses for both installation and maintenance. Second, heat and smoke do not spread

instantaneously, making the sensors intrinsically slow at picking up these signals from the environment. To overcome these challenges, computer vision-based fire detection techniques have been gaining traction over the last couple of years. Cutting-edge computer vision algorithms utilizing layers of pattern recognition can make feasible early detection of fire. For accurate detection, these layers attempt to analyse various distinctive features of fire, which include its characteristic color, shape, flickering rate, and growth, and the spatiotemporal difference in frame sequences. This paper presents a method of (Luminance (Y), Blue-luminance (U), Redluminance (V)) (YUV) color segmentation, a generic shape analysis technique, and enhanced optical flow analysis of candidate fire regions for robust detection of fire. The remainder of this paper is organized as follows.

Proposed Model

The proposed model utilizes three major phases for the detection of fire.

Chromatic Segmentation in YUV Color Space

The non-chemical fire has a characteristic color that almost always appears in the red-yellow range. Chromatic segmentation based on this distinctive feature thus naturally acts as the primary trigger for fire detection. Among the existing segmentation methods, many are well-formulated using principled techniques of image processing, while others are based on heuristic thresholds and empirical observation. Based on the experimental analysis, pixels in most flames exhibit a Red channel value greater than that of Green, as well as a Green channel value greater than that of Blue: Where $R(i,j)$, $G(i,j)$, and $B(i,j)$ are the Red, Green, and Blue (RGB) channel values of pixel (i,j) respectively. We have used YUV color space as it considers the chrominance and luminance of a pixel. The constraints in can be equivalently expressed in the YUV plane by converting from RGB to YUV using the well-known conversion rules specified in. Based on the converted output, the following rules can be used to define a fire pixel, so that we obtain the following generic pixel. A common limitation of chromatic filtering algorithms is that they perform logical conjunction of all the rules to segment out the region of interest. This approach often fails to detect visually distant flames. To overcome this limitation, a more liberal approach is undertaken in the proposed system by taking a vote on all the rules and segmenting the image

based on the majority decision, as described in Equation. The smaller is the value for M in Equation, the larger is the number of false positives. The value is therefore chosen such that any potential fire pixel is not ignored by being too restrictive. By empirical study, in this paper, the value of M was adjusted to be 4. Additionally, the shape and optical flow analyses are designed to address this issue. that the majority decision rule can extract fire regions more accurately.

Shape Analysis

A distinguishing feature of flames is their complex shape that varies rapidly as a function of time. As suggested in, blobs from two consecutive frames from the previous step are identified, and the ratio of their perimeter and area is calculated. Most of the non-growing regions were eliminated due to the shape and growth analyses.

Optical Flow Analysis

Fire is a turbulent physical phenomenon, and the inner edges of fire plumes exhibit a chaotic displacement as a function of time. This feature is so unique to fire that successful extraction results in a dramatic reduction of false positives. This feature can be extracted using optical flow tracking algorithms. In the proposed system, an enhanced Lucas-Kanade optical flow analysis technique is suggested to calculate the turbulence of fire. The Lucas-Kanade Tracking (LKT) algorithm uses a set of corners to track a sequence of frames. At first, the Features from the Accelerated Segment Test (FAST) algorithm are utilized to detect 50 corners in the firelike regions. These corners are subsequently fed into the Lucas-Kanade tracker to be tracked across a sequence of frames. The LKT algorithm then calculates and returns a set of motion velocity vectors for each of these corners from frame to frame $t+1$. The result of the LKT algorithm on a set of corners is depicted. The velocity vectors of the corners detected in fire are represented using purple lines. The longer lines indicate faster movement of the corner. This flow rate complexity represents the chaotic, non-deterministic motion of the local features of fire. Consequently, the value of this will differ between fire and non-fire objects. The proposed model makes use of this average flow rate complexity to analyze the possibility of fire. In the empirical study, the value of in Equation has been concretized to be 10, and the complexity is calculated n times. Every time the complexity exceeds the threshold 0.5, a counter is

incremented. If the complexity exceeds the threshold more times than not in n attempts, the scene is considered a potential fire scene. The average flow rate values for 1000 such n -frame scenes are then extracted and subsequently trained using a scaled conjugate gradient back-propagation neural network.

- a) The original frame from a video with sunlight shimmering through a forest containing trees and leaves having the characteristic color of fire.
- b) The corresponding motion vectors obtained from. Motion vectors of a non-fire scene with fire-like properties. To test the efficacy of the model, the n -frames scene is fed to the neural network for analysis, which then attempts to classify the n -frames scene as a fire or non-fire scene. The model successfully demonstrates promising performance in picking up a pattern from the flow rate complexity. illustrates the velocity vectors of a sample fire, where the vectors of fire are quite dispersing and chaotic. The turbulent motion vectors in the fire scenes depicted in are easily distinguishable from the relatively calm motion vector in the non-fire scene.

Experimental Results

To evaluate the performance of the proposed model as a simulation tool, we utilized MATLAB and tested it using a diverse dataset containing videos in a wide variety of scenarios. The dataset includes videos containing the fire at different illumination levels and videos with ordinary objects that display fire-like color. The dataset was prepared predominantly using videos from the Miviva dataset with some videos from Zenodo, and the rest were taken from YouTube or recorded personally. The videos can be broadly divided into three categories: indoor, outdoor, and forest fires.

The proposed model yields a comparatively greater true positive rate in scenarios where the fire is situated in a distant region. It excels over the competing models in these cases because these models employ a stricter set of color segmentation rules in their first phase of analysis. The restrictive color rules fail to extract the small fire regions in the scenarios where the fire is located far distant. Consequently, the comparatively liberal color rules employed in the proposed model yield a better true positive rate. The results of the enhanced segmentation rules that yield this better accuracy of True Positive (TP) in percentage (%). The average TP rate is the percentage of fire pixels correctly identified in a scene, whereas the average

False Positive (FP) rate is the percentage of non-fire pixels misidentified as fire. Enhanced Equations and identify a greater number of true fire pixels compared to Equation. This test was carried out against frames containing very bright flames. Even better performance is provided by the majority voting rule in Equation (13), which identifies 98.9% of fire pixels correctly. This test was carried out against frames containing flames situated far away. Although it simultaneously increases the average FP rate, the effect of this is eliminated by the optical flow analysis phase. Additionally, the performance of the proposed model with enhanced color rules and an optical flow analysis technique is compared with the model presented by and Assessment is performed concerning the rates of true positives and false positives in a scene of 10 frames, and the corresponding model's capability to correctly classify the 10-frame scene as a fire scene or non-fire scene. the comparison of the models in different environments, which reveals that the proposed model yields an overall average accuracy of 97.2%, outperforming both models. Furthermore, the proposed model achieved greater success in the reduction of false positives, demonstrating an only 2.21% false-positive rate compared to 7.78% and 6.76% for models, respectively. model is typically weaker when there is sunlight in the background. uses a novel foreground extraction technique that exploits the flickering property of fire by detecting the rapidly changing blue channel value of fire pixels. However, this property is also true for the pixels in sunlight, yielding an alarming rate of false positives in scenes containing sunlight. The proposed model, in contrast, does not suffer from this weakness due to its reliance on dynamic optical flow analysis, which yields different results for fire and sunlight. Similar to the proposed model, model also does not suffer from the limitation of Khan's model. However, model is weaker when objects exhibiting the characteristic color of fire are moved in a haphazard, unpredictable way. employs an optical flow analysis technique that calculates the variation of optical flow over time. However, the model uses a naïve thresholding approach that fails to differentiate the unpredictable motion of fire from the unpredictable motion of an object moving in a non-deterministic way. Alternatively, the proposed model calculates the average flow rate of candidate pixels and employs a back-propagating neural network to identify a pattern from the average flow rate, which reduces the false positives rate. Overall, the model is powerful in its ability to detect fire in all types of scenarios. It performs equally well in detecting indoor, outdoor, and forest fires at different levels of illumination and varying levels of

distance from the camera. Moreover, compared to the competing models, it increases the true positive rate of detection and significantly reduces the false positive rate.

Conclusions and Future Work

A robust fire detection system has been proposed in this paper which analyzes several properties of fire, including color, shape, and turbulence. An enhanced chromatic segmentation algorithm using a majority voting system to detect $F_{candidate}(x,y)$ if (4) has been adopted in cohesion with a generic shape analysis technique to detect two of the most fundamental visual properties of fire. As the final and most important step of the model, an enhanced LKT optical flow analysis algorithm was proposed to detect the disorderly motion of fire plumes. Otherwise, the rate of false-positive would be higher. The proposed model was tested using a diverse dataset, and it outperformed existing models by exhibiting an average accuracy of 97.2%. Future work will be intended to carry out the optical flow analysis step using a hybrid neural network that is trained directly with a set of principled turbulence values as the feature vector.

2.3 Machine vision-based real-time early flame and smoke detection

Abstract

This paper proposes a novel real-time machine video-based flame and smoke detection method that can be incorporated with a surveillance system for early alerts. Automatic monitoring systems use the motion history detection algorithm to register the possible flame and smoke position in a video and then analyze the spectral, spatial and temporal characteristics of the flame and smoke regions in the image sequences. The spectral probability density is represented by comparing the flame and smoke color histogram model, where HSI color spaces are used. The spatial probability density is represented by computing the flame and smoke turbulent phenomena with the relation of perimeter and area. Statistical distribution of

the spectral and spatial probability density is weighted with the fuzzy reasoning system to give the potential flame and smoke candidate region. The temporal probability density is represented by extracting the flickering area with level crossing and separating the alias objects from the flame and smoke region. Then, the continuously adaptive mean shift (CAMSHIFT) vision tracking algorithm is employed to provide feedback of the flame and smoke real-time position at a high frame rate. Experimental results under a variety of conditions show that the proposed method is capable of detecting flame and smoke reliably.

Introduction

Video surveillance can be used to monitor fires and burn disasters. Traditional fire protection methods use mechanical devices or humans to monitor the surroundings. The most frequently used flame and smoke detection techniques are usually based on particle sampling, temperature sampling and air transparency testing. An alarm is not raised unless the particles reach the sensors and activate them. Further, traditional detectors seldom provide additional descriptive information about the flame location, size, burning degree and so on. Surveillance systems require a large number of people to watch the monitor screens all day long, and hence the disadvantages are the increased possibility of human errors, the amount of data required to be stored and the high cost. In, light section image detection is proposed to overcome the shortcomings of conventional beam-type smoke sensors and detect the early fire. However, the drawback of this method is that the detection system requires many elements such as infrared radiation arrays, infrared cameras and a signalprocessing unit to be set up. Moreover, such systems are limited to monitoring only the fire smoke flow. 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. Fire is a phenomenon created by the eruption of glowing combustion products in turbulent motion. Fire has unique visual signatures. Color, geometry and motion of fire regions are all essential for recognition. Fire is a complex natural phenomenon involving chemical reactions and turbulent flow. The turbulent flames due to an uncontrolled fire expose a flickering characteristic. Fire flickering results from the chemical nature of fire. Diffusion flames flicker due to local and global extinction resulting from non-availability of oxidizer. Global extinction is identified as the stage during combustion when the overall reaction progress is not sufficient to sustain the flame. After global extinction, the diffusion of fuel and oxidizer continues until the conditions for combustion are again met and a flame reappears. Therefore, the actual moment when no flame exists is not actually perceived, but rather a flicker in the flame is observed. The entire process occurs in a fraction of a second. Another feature of fires is that the shapes are changeable because airflow caused by wind will make the flames oscillate or move suddenly. As a result, smoke boundaries also oscillate with a low frequency during the early stages of fire. Flames are always pointed at the top and spread upward; this is also an important visual feature of fire, since the hot gases in the flame are much hotter and less dense than the surrounding air, so they move upward toward lower pressure. Smoke is a compound of hydrogen, carbon and oxygen. The constituents and quantity of the smoke depend on the chemical components of the combustible material, the burning temperature, the supply of oxygen and so on. The visual pattern of smoke is difficult to model, and the smoke density varies with the surroundings. However, smoke from an uncontrolled fire can be easily observed even if the flames are not visible, and the fire can be detected early on before it spreads around. Flame and smoke detection methods via video processing and the related research work can be found presented a fire detection algorithm using color video input with a pre-partitioning scheme under some restricted conditions and without rejecting the similar fire-like alias. Computer vision studies by use spectral analysis and flicker of fire as a property to recognize existence of fire in a scene. In, hue and saturation are adopted as feature vectors to extract the fire pixels from visual images. In fire flame features based on the HSI color model are extracted, and regions

with fire-like colors are roughly separated from an image by the color separation method. Then, the image difference method based on chromatics is used to remove spurious fire-like regions such as objects with similar fire colors or areas reflected from fire flames. In the fuzzy-based dominant flame color lookup table is created, and fire regions are automatically selected. However, either the solution does not consider the temporal variation of flames, or the approach is too complicated to process in real-time. In, autoregressive (AR) and temporal models of fire regions in visual image sequences are presented, and the shapes of fire regions are recognized in Fourier coefficients. Since Fourier transform does not carry any time information, FFTs have to be computed in windows of data, and the temporal window size is very important for detection. In, in addition to fire color clues, the flame flicker process is detected by using a hidden Markov model and the transition probabilities for flame and non-flame models must be calculated off-line first. proposed a real-time algorithm for fire detection in video sequences .They combined motion and color clues with fire flicker analysis by adopting the wavelet transform to extract the frequency feature of the central point of the flame and using it as the fire-detection feature. However, the false alarm problem is caused by variations in light. In , a two-class support vector machine (SVM) classifier is used for the final fire-pixel verification and false alarm reduction. However, additional computation time is required for the SVM. In, image features such as motion, flickering, edge blurring to segment moving and edge-blurring regions were extracted from video to detect smoke. The methods used to extract these features were background subtraction, temporal wavelet transformation and spatial wavelet transformation. In, smoke pixel was judged with a chromaticity-based static decision rule and a diffusion-based dynamic characteristic decision rule. In [16], a video smoke detection technique is proposed comprising background subtraction, flickering extraction, contour initialization and contour classification using both heuristic and empirical knowledge of smoke. In, to avoid the false flame detection due to the interference of background illumination and color such as neon or traffic light, the time-varying property of flame geometry is taken into account. To achieve fully automatic surveillance of fire detection, a novel visual real-time flame and smoke detection method is proposed on the basis of color video processing to meet the above requirements. The proposed real-time fire flame and smoke detection technique employs spectral features (i.e., saturation and chromatic features),

spatial features (i.e., disordered features) and temporal features (i.e., flickering feature) for extracting flame pixels and smoke pixels. Fire detection is first derived by the chromatic feature of flame and smoke. With regard to the general color of flame and smoke, the proposed system acknowledges that the flame and smoke chromatic feature can dominate the decision function of flame and smoke pixels. With regard to the spatial features of flame and smoke, the flame and smoke will flicker suddenly, which implies that the flame and smoke shape is dynamically changeable in visual images. Thus, the extracted flame and smoke pixels will be further verified through a check of the disordered characteristics of the burning fire. Statistical distribution of the chromatic and disordered measurement value is weighted with the fuzzy reasoning system to give the potential flame and smoke candidate region. The temporal probability density is applied by extracting the flickering area with level crossing to separate the alias objects with the flame and smoke region. Then, the continuously adaptive mean shift (CAMSHIFT) vision-tracking algorithm is employed to provide feedback of the flame and smoke real-time position. The main differences and advantages of the proposed approach are as follows.

- It combines the spectral, spatial and temporal flame and smoke characteristics to perform the machine visionbased fire early detection and tracks the flame and smoke region based on the CAMSHIFT algorithm.
- The image sequences are processed and the fire flame and smoke are recognized at 20[ms] or lower video rates by applying the detection and tracking algorithm. After flame or smoke is detected, only limited regions surrounding the current tracking window need to be processed, and hence there tends to be a substantial reduction in computational costs.
- The proposed flame and smoke detection algorithm not only has a high correct decision rate and better capability to avoid false alarms caused by environmental illumination, but it can also track the flame and smoke regions from the image much more precisely than the other systems.
- Multimedia timer functions and PC-based real-time control are used along with specific software modules written in C++. Highly flexible multiple threads for the detection and tracking process are assigned by the multimedia timer scheduling thread software. compares the features of the existing flame and smoke detecting techniques that have already been investigated in the literature. This paper is

organized as follows. The overall flame and smoke detecting algorithm structure is introduced and analyzed. Then some experimental results are given. Finally, the conclusions are given in

Flame and smoke detection algorithm

The proposed flame and smoke detection algorithm consists of four steps:

- (1) moving pixels or regions in the current frame of a video are determined with the motion history image (MHI);
- (2) the HSI colors of moving pixels are checked;
- (3) if the histogram of moving pixels is correlated with the flame and smoke color histogram, then the disordered measurement and temporal analysis are performed to determine if flame and smoke colored pixels flicker or not; and
- (4) CAMSHIFT is applied to track the flame and smoke region.

Moving motion segmentation with motion history image (MHI)

The region of interested moving pixels represents motion in successively layered image differences that create residual silhouettes of moving objects by using the MHI where successive layering of image silhouettes is used to represent the patterns of motions. The MHI is a scalarvalued image where intensity is a function of recency of motion. This moving history representation can be used to determine the current movement of the object and to segment and measure the motions induced by the object (e.g., fire flame and smoke) in a video scene. MHI representations have the following advantages: a range of times from frame to frame to several seconds may be encoded in a single image, direct recognition of the motion itself is possible, motion recognition is not computationally taxing and real-time implementation is possible, and the motion within the detecting scene can be monitored. An MHI is used to represent how the motion of flame and smoke is moving, since the outward boundaries of flame and smoke are less prone to mis-detection than the source regions of flame and smoke. In an MHI, pixel intensity is a function of the motion history at that location, where brighter values correspond to more recent motion. It should be noted that the final motion locations appear brighter in the MHIs. The MHI_{τ} , where pixel intensity is a function of the temporal history of motion at that point The result is a scalar-valued image where more recently moving pixels are

brighter and implicitly represent the direction of movement. Hence, the motion gradient and segmentation are also acquired by the MHI algorithm and the orientation and direction of the object can be computed.

Spectral characteristics correlation

To detect possible flame and smoke pixel candidates, the first step is to transform the color space into HSI color space and analysis can be performed. Hue is the dominant color (red, green and blue) of an area, and saturation is the colorfulness of an area in proportion to its brightness. Intensity is related to the color luminance, e.g. human skin occupies a small portion of the H and S spaces. The advantages of the HSI space are intuitiveness of the components and explicit discrimination between luminance and chrominance. The hue, saturation and intensity components of the HSI model are normalized into the following ranges: 0° hue 360° , 0 saturation 255 and 0 intensity 255. These important facts make the HSI color model useful to simulate the color sensing properties of the human visual system. This is because the hue and saturation components are intimately related to the way in which human beings perceive color. The HSI color system that corresponds to projecting the standard red–green–blue (RGB color model) color space along its principle diagonal from white to black is applied to avoid the influence of lighting changes. The HSI color histogram of moving regions is used to compute the correlation with a predetermined flame and smoke histogram distribution, which represents the possible flame-pixel or smoke-pixel spectrum in the video in HSI color space. To detect possible flame-pixel candidates, the fixed model template of fire flame spectral histogram, which is based on empirical analysis results for the fire flames with colors from red to yellow and was created to detect the flame-colored pixels, is applied. Since hue is an attribute of the pure color of the image scene, it was demonstrated that it can be used in assessing the prospect of numerical labeling of the flame colors. Hence, the detection of flame pixel is carried out using the hue channel histogram correlation analysis with the fire flame template, which maps the hue value of general flames to be distributed from 0° to 60° . To detect possible smoke-pixel candidates, the smoke histogram template is obtained from sample images containing smoke regions. The detection of smoke pixels is carried out using the saturation channel histogram correlation analysis with the smoke template, which maps the saturation value of general smoke to be distributed from 0

to 40. According to, smoke visibility refers to the attenuation of light or opacity along the line of sight, and the saturation will decrease when opacity increases. No matter what the smoke color is, the gray level histogram becomes concentrated if the smoke appears.

Chaotic spatial structure analysis

Some flame-like and smoke-like regions in an image may have the same colors as flame and smoke, and these flamesimilar and smoke-similar areas are usually extracted as the real flame and smoke from an image. To validate a real burning fire, in addition to using chromatics, spatial features are usually adopted to distinguish other fire flame aliases. Such fire flame dynamics include sudden movements of flames, changeable shapes, growing rate and oscillation (or vibrations) in the infrared response. Smoke and flames are both turbulent phenomena. In the case of turbulent flow the chaotic nature of the flow is an important feature. If the contours of an object exhibit rapid time-varying behavior, then this is an important sign of the presence of flames or smoke in the scene. The shape complexity of turbulent phenomena may be characterized by a dimensionless edge/area or surface/volume measure. The moving object regions with the disordered ratio of perimeter to area for the extracted flame

As the complexity of a shape increases (i.e., the perimeter increases with respect to the area), the value associated with the disordered ratio also increases. The chaotic and turbulent nature of a region can be detected by relating the extracted spatial features to the flame and smoke likelihood region. The likelihood that a flame-like region is a flame region is highly correlated with the parameter .

Temporal analysis

To detect flame and smoke correctly based on the color information is not always enough. There are lots of objects which expose a similar color property to the flame and smoke spectrum. The key to distinguishing between flame and flamecolored objects and smoke and smoke-colored objects is the nature of their motion. The flames in fire dance around, so any particular pixel will only see fire for a fraction of the time. This kind of temporal periodicity is commonly known as flickering. The flicker of fire flame causes the spectral values in the fire

flame region to fluctuate in time. The flicker in smoke is also used as additional information. The candidate regions are checked whether they continuously appear and disappear over time. To distinguish flame-like and smoke-like aliases, the temporal variation of flame and smoke color regions is applied. Temporal variation in conjunction with flame and smoke spectral histogram $Scorr$ and disordered ratio is employed to detect flame and smoke regions. A fuzzy-logic-enhanced approach is used to decide if flame and smoke are being detected and could spread to cause an accident. The single output decision quantity is used to give a better likelihood that a pixel is a flame or smoke pixel. In this work, the two main input variables for the fuzzy logic are the chromatic histogram correlation $Scorr$ and disordered ratio. There are scaling constants G_s and G using which the values of G_sScorr and G are rescaled to fit the range. Then, the values of G_sScorr and G are fed into a fuzzy system to give flame– smoke likelihood, i.e. flame and smoke index u information from the output. Both G_sScorr and G values comprise four fuzzy regions: ZE (zero), PS (positive small), PM (positive middle) and PL (positive large). For simplicity, four standard triangular membership functions for the fuzzy region variables $\{ZE, PS, PM, PL\}$ are used here. Then, the minimum implication operation is introduced into the fuzzy inference engine to evaluate the decision rules stored in the .The columns and lines correspond to the chromatic histogram correlation and disordered ratio score (inputs of the fuzzy reasoning system); the values of the matrix correspond to a measure index that shows how likely it is that a region located at a spatial location belongs to the flame and smoke pixel (output of the fuzzy reasoning system). The fuzzy output consists of four singletons $\{\mu_1, \mu_2, \mu_3, \mu_4\}$. Weighted average defuzzification is applied on the union of all rule outputs in order to find a flame and smoke index μ . In weighted average defuzzification, the output value u is the geometrical center based on the above fuzzy rules. With the values of the four singletons in hand, the fuzzy aggregated.

CAMSHIFT tracking algorithm

The CAMSHIFT algorithm is a non-parametric technique that can track a specified target's 2D position efficiently across a series of images. The CAMSHIFT tracking engine is based on the histogram projection algorithm that is a useful technique

for color object recognition, especially for object identification in complex background surroundings. Histogram backprojection is a primitive operation that finds and identifies the association between pixel values in a grabbed image and the values in a particular histogram bin. Histogram and back-projection performed on any consecutive frame would generate a probability image on which the value of each pixel represents the probability of the exactly same pixel from the input belonging to the target histogram that was used. Given that m histogram bins are used, we can define n image pixel locations. That is, values in the histogram bin, originally in the range $[m, \max(y \wedge u)]$, now lie in the new range. In the end, the input pixels with the highest probability of being in the sample histogram will be mapped onto a 2D histogram back-projection image with the highest visible intensities. The CAMSHIFT algorithm can be summarized. The CAMSHIFT algorithm is a non-parametric technique that can track a specified target's 2D position efficiently across a series of images. When tracking the 2D position of a colored object, in our case a flame and smoke colored region, the CAMSHIFT operates on a color probability distribution image derived from color histograms. The center and size of the targeted object region are computed and used as settings for the search window on the following frame of the video sequence. the images of the tracked object in the digital pictures that have been recognized and processed, and the yellow bounding ellipse presents the smoke region tracked by the CAMSHIFT algorithm. The tracking routines can be done within 20[ms] video rates by applying the CAMSHIFT algorithm. The calculation of the color probability distribution is not performed on the entire image, but only on limited regions surrounding the current CAMSHIFT window, which includes images of the specified object that are transformed into a discrete probability image. This tends to result in a large reduction in the computational costs.

Analysis and comparison

The analysis of experiments implementing the proposed process derived in previous sections is presented in this section. To realize the proposed flame and smoke detection system, a general digital color video camera is used to capture several flame and smoke sample image sequences with the format of pixel resolutions of 320×240 . The process rate of the system performs at the rate of 11–57 frames s⁻¹ on an Intel Pentium computer running at 2.0 GHz. For the detailed evaluation, fuels were burned in different situations and the decision rules on spectral, spatial and temporal features described previously were used. the flame candidate is detected at the 129th frame and the smoke candidate is detected at the 1449th frame; the flame begins to be tracked with CAMSHIFT and the fire alarm is raised from the 130th frame, and a smoke-alarm at the 1450th frame because the smoke spreads more than the flame and thereby the smoke dominates the alarm. Results of implementing the proposed flame and smoke detection system in various situations of illumination (movies 1–8), plus the additional test data (movies 9–12) smoke and flame detection results, where the field nt is the number of frames of a video clip. The field f^- is the number of false negative flame and smoke, which means that the system does not detect flames and smoke in an image frame while there is indeed flame and smoke frames in it. Similarly, the field f^+ is the number of false positive flame and smoke, which means that the system recognizes flame and smoke in an image frame while there is no flame and smoke at all.

Conclusions

In this paper, an approach to detect the flame and smoke of real-time alarm systems is presented. Spectral, spatial and temporal features and fuzzy logic are employed to extract real flame and smoke and are adopted for helping the validation of that flame and smoke. The experimental results show that flame and smoke can be successfully detected under various environmental conditions, i.e. indoor, outdoor, day, simple or complex background image, etc. The proposed algorithm can be integrated into the existing surveillance systems to achieve detection of flame and smoke in video databases as well as real-time detection of fire. In future work, the neural network will be applied to train the LCR threshold for increasing the reliability in complex moving environments. In addition, multiple fires may break out

simultaneously, and hence we conduct research on tracking multiple fire regions concurrently.

2.4 Machine Learning Based Early Fire Detection System using a Low-Cost Drone

Abstract:

This paper proposes a new machine learning based system for forest fire earlier detection in a low-cost and accurate manner. Accordingly, it is aimed to bring a new and definite perspective to visual detection in forest fires. A drone is constructed for this purpose. The microcontroller in the system has been programmed by training with deep learning methods, and the unmanned aerial vehicle has been given the ability to recognize the smoke, the earliest sign of fire detection. The common problem in the prevalent algorithms used in fire detection is the high false alarm and overlook rates. Confirming the result obtained from the visualization with an additional supervision stage will increase the reliability of the system as well as guarantee the accuracy of the result. Due to the mobile vision ability of the unmanned aerial vehicle, the data can be controlled from any point of view clearly and continuously. System performance are validated by conducting experiments in both simulation and physical environments.

Introduction

The perspective based on visual indications relating to early detection systems in forest fires. In this period where the number of systems developed by utilizing unmanned aerial technology is increasing day by day, unmanned aerial vehicles will be used to achieve the targets of minimizing the destruction of our forests which are the lungs of the world and optimizing the usage of workforce and time resources. As a result of the application carried out in line with the subject of this paper, it is proposed that the system based on the detection of smoke image with unmanned aerial vehicle can provide a great benefit in reducing the error rate occurring in fire detection. The microprocessor in the system has been trained with deep learning methods and has been given the ability to recognize smoke image, which is the earliest sign of fire

diagnosis. The most fundamental problem in the common algorithms used in fire detection is the high level of false alarm and overlook rate. Confirming the result obtained from the detection and defining an additional proof will increase the reliability of the system as well as the accuracy. Since the drones provide a mobile vision, the point of view can be controlled by the ground station can manipulate it for the sake of the accuracy of the result. The application developed in line with the subject of the paper was implemented in both simulation and physical environments and the advantages of early fire detection system and analysis results are discussed in the conclusion section of the article.

Motivation

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. In this way, it can be easily separated from other natural components in the sky and forest by means of various filtering, edge detection and color recognition tools. However, it is too late to intervene when the fire and flame become visible. For this reason, based on the smoke data, which is the earliest sign of the fire, convincing the fire diagnosis will pave the way for a faster reaction. Nevertheless, the results obtained by processing image or video input by image processing algorithms in today's developed applications contain high errors. In the event of a fire, no signal is given or the system goes into an alarm state when there is no fire. These conditions should be minimized for early and on-site intervention. At this stage, a control system patrolling with unmanned aerial vehicles will be functional in terms of providing fire and providing a clearer view. The choice of a microprocessor controlled by an electrically powered microprocessor from the power component of unmanned aerial vehicles will save time in the assessment of the possibility of fire using the "TensorFlow library" of the microprocessor in the vehicle without the data received from the vehicle being sent to the control station for artificial smoke detection. The main objective of this study is to minimize the rate of false reports and omissions and to optimize the process. Designed Drone is illustrated As well as Bottom and Top body designs of the drone is given in Appendixes A and B Overall involves materials and method whereas will followed by the experimental results section and the paper is finally concluded.

Materials and Methods

Forest fire is a natural disaster and its havoc can be significantly reduced in case of early detection and early intervention. Despite the superior efforts of the firefighters, massive diffusion of fire is unavoidable for some reasons such as traffic, late or false alarms, in case of where resides hard to intervention. In the case of forest fires the location and the time required to reach are the most important obstacle to be overcome. Because forest fires have the fastest spreading speed and the most noticeable type of fire. Due to the fact that forest fires that occur frequently, green areas in our world are getting smaller each day. Forest fires have been intensified in some provinces. It is known that helicopters are involved in firefighting besides water tankers and pumper vehicles and so faster results have been obtained from land vehicles. The objective of this study is to minimize the rate of error in the results obtained in detecting the forest fires by the ability of the unmanned aerial vehicles to view and move. The Raspberry Pi device is a powerful and high capacity device used in many robotic and IoT projects, offering the programmer a mobile production environment. The first step in making the Raspberry Pi minicomputer usable with unmanned aerial vehicles will be to connect the device to the system, which can be fed from the internal battery of the unmanned aerial vehicle. Otherwise, other methods that can be used, such as feeding the device with a powerbank to be fixed on the unmanned aerial vehicle, are not a desired feature since it will increase the weight of the vehicle. The current and voltage values of the electricity required from the LiPo battery of the Raspberry Pi are too high that it prevents feeding the device from the vehicle with trip wiring. For Raspberry, this regulator is placed between the power distributor and the inlet slot to overcome this obstacle. A converter battery adjuster circuit for this, whereas, the black pin end pin ends of the BEC circuit connect to the ground (ground) end of the power distributor, the red pin end of the circuit to the voltage-end of the distributor, and the triple end of the rechargeable circuit to the Raspberry Pi 2.4 and 6 function pin inputs. While designing a frame for the four-engine rotary wing unmanned aerial vehicle to be used in the project, a perforated model made of carbon fiber material was preferred by considering the aerodynamic effect and the center of gravity distribution. When looking at the drone designs, the reason for choosing the H-type frame can be seen as the potential of the

drones with this type of frame to perform a more robust roll while performing load-bearing tasks due to the reliable mass base and arms extending apart from the frame. Htype drones can recover more easily after taking sharp bends, and it can be predicted that the battery placed in the center of the hull will last longer as it will take less damage in possible drops. However, in the original design of the vehicle used in the project, it is planned to place electronic components in the sandwich model on the coreless body. Thus, since the electronics and battery will be located in a sheltered compartment during trials and flight training, the second of the two main differences between the X and H type cases remains ineffective. The fastest move to be made during the fulfillment of the project requests will be the movement to the left and right. Considering this requirement, the most obvious difference that makes the H frame reasonable when the X and Y frames are compared will also lose its importance. Generally, type X frame is used in racing drones. Considering that the fast course of the vehicle to follow will provide a wider control, the advantages of using the X-type drone in racing vehicles played a role in the choice of the X-type body in the body design of the project vehicle. These advantages are; more balanced distribution of the center of gravity, the mechanisms involved in drone moves are in a symmetrical co-operating principle, and the pilot can be more easily orientated to the vehicle control commands in control and control. In the project vehicle, the distance of the engines from the body / center arm length was designed as 30 cm. The four engine rotary wing unmanned aerial vehicles have a well-established place with distinct advantages among lightweight multicopters. The most important advantage is that the unstable drone design minimizes the balance handicap due to the motor / wing symmetry compared to the fixed wings. Each engine to be added will add weight to the design, but at the same time, since the increase in engine power will directly increase the handling, acceleration capacity (torque) and performance of the drone, the four-engine design will balance this contrast well, producing the most appropriate results. The electronic speed control unit, engine and propeller components make the drone propulsion system. Speed control in brushless motors can be done with constant torque, since there is no friction, the motor; it does not arc, dust, warms up and works at high efficiency. In the design of the vehicle, it was found appropriate to use four 770 KV short shaft avroto brushless motors. In each engine, four propellers made of high-strength plastic / nylon or carbon fiber nylon material will be used, taking into account the estimated weight of the project vehicle and the risks that may

arise in possible collision. Instead of 3 blades of the propeller to be selected, it has been determined as 2 blades, inclination angle of 4.5° (11.43 cm) and propeller diameter of 10" (25.4 cm). This selection was based on peer racing drones. When calculating the thrust value of the unmanned aircraft system, d symbolizes the propeller diameter in inches and is included in the formula as 10" (25.4 cm), which is a known variable. RPM value represents the numerical value of the propeller in one minute. The pitch value represents the value of the road traveled in a propeller lap in inches, and V_0 represents the forward propeller speed in m / s unit. In the forward dynamic propulsion equation, the test data of the Avroto M2814-11S engines used in the vehicle was used when calculating the thrust value [11]. The fact that the vehicle has a unique design is one of the main objectives and achievements in the project. In order to carry the sub-systems of the vehicle (Raspberry Pi, Camera Module, Landing Gear, Battery, 4 ESC etc.) in a balanced and sheltered way, the model formed by the parallel placement of two empty plates called the sandwich model has been found appropriate. Taking into account the aerodynamic elements, the sandwich model of the vehicle is supported by the perforated body design. After the vehicle was created, flight attempts were made in two ways by mounting the battery above and below the case. The working principle of the load carrying mechanism in unmanned aerial vehicles with rotary wings is based on the force of the air produced by the thrust units downward. Thus, any design in which propellers interact with each other or another object negatively affects the stability and performance of the vehicle. After this was observed in the test results, the batteries were placed under the body. During flight tests, no losses were made except for the propeller.

Software Design

Raspbian operating system is installed on the device, which has been developed and can run many programs smoothly and quickly. OpenCV is suitable for the use of image processing and also installed into the device. Besides, Tensor Flow library, a leading machine learning library, is also installed for object detection problem. It is possible to install and make programs with standard Linux commands through the terminal command line after the installation of the operating system on the microcontroller. Python 3.6 language is utilized to bring artificial learning skills into

the tool. In the model used to distinguish the smoke image from the background elements, several post processing steps are carried out involving, background separation, color and edge modeling, and data normalization. With the help of built-in methods, functions and defined values in the libraries used in this steps, the steps to be taken during the calling and use of the model after the training phase are carried out in a cost-free manner and in order to prevent the processor from being exhausted. In the subheadings of the Materials and Methods section, the installation and start-up steps of the application, the identification and training of the model, and the process of installing the model on a newly purchased Raspberry are included. As aforementioned above, Raspberry Pi is supported by OpenCV and Tensorflow libraries, integrated with a self-made drone so as to detect forest fires. Tensorflow is a library used to train the system for the object to be distinguished in object recognition applications. In the Tensorflow model, the distinguishing properties of the object are extracted using a data type called tensor. In terms of the Tensorflow library, tensor is a multidimensional data structure consisting of primitive (integer, float) data contained within a multidimensional array. It serves to keep the difference and similarity mathematically determined when certain regions of the images in the stream are specified to contain the same object. The Tensorflow diagram aims to identify the factors that are effective in the most accurate classification of test images and to produce a property map for the objects in the model, with improvements and training phases. When the object recognition algorithm is running, a series of computational operations, which are made up of nodes and can be represented by a flow diagram, are executed when the object to be distinguished is identified in the model. Generally, each node takes a tensor, calculates it and outputs a tensor. In learning models, it is aimed to get different and improved outputs with the same inputs. In order to achieve this, tensors in the flow are updated and iteration is maintained by using Tensorflow variables, operations, placeholders and constants. Improvement updates in machine learning cannot be done manually, there is an API called loss function, which will minimize the instantaneous value of the function used to measure the difference between the estimated success value of the model and the success of correlation value is fed to that object as a similar relationship is encountered in the region containing the object in the other pictures. The training pictures in the batch, the specific iteration unit, are finally compared with the test data

and the model is updated and optimized according to the difference between the classification and the class to which it actually belongs. The statistics and results of the mathematical evaluations that the trained system is trained to give fast and low error rates are given in this section. The ssdlite mobilenet model, which is a sub-model of the supervised deep learning school called Model Zoo, which contains pre-educated models in the Tensorflow library, is taken as an example for training in this study. There is a speed-precision swap that creates a kind of balance between sub-models. The criterion to be considered when choosing a model is the purpose for which the model will be used. Because in programming there is always a trade-off balance that makes it necessary to waive certain things while improving certain things. In the real-time object recognition application, the speed of the model is in the foreground, while the accuracy rate remains in the background. Essentially, the models, trained with COCO training set, were investigated. According to which, the “ssdlite mobilenet” model, most likely to give the best results when working on real time smoke data on Raspberry, is adapted for this study. An example smoke detection within the developed software system. For the training process, 280 images, representing smoke, most of those obtained from google whereas some of them are generated in physical environments. Consequently, those data are employed the training phase of the model .

Results

When evaluating the loss graphs, it would not be correct to call the model with the lowest loss value that we obtained the most successful model. This can only show that that model can be trained more quickly than other models. The training was ended in different models and different iterations in each model. For instance, illustrates a model employs %90 of data for training. Comparing the results to the point where each model reaches the optimum level, comparing the results and comparing the loss values at the end of the trainings can be considered as a relatively accurate omparison parameter. There are two versions of Tensorflow running on CPU and GPU. The model is trained in two ways. The same model working on the same data was able to reach the point where the training process carried out on the CPU came within 10 hours 32 minutes 54 seconds, on the GPU in 1 hour 39 minutes 47 seconds. In this case, it was observed that the model was trained 6.34 times faster compared to the

CPU when the deep learning algorithms carried out on visual data were carried out on the GPU. A number of analysis procedures were carried out on four models run on the same control group to choose between the different percentage of training and test data dropped and four different models trained. Estimation success of the models in different respects was measured and models were compared to each other and one of them was preferred to be used. There are four models are defined based on the distribution of the train and test data.

There are some standard evaluation parameters used to measure the overall performance of deep learning models. These are total accuracy, positive interpretation power, negative interpretation power, sensitivity, F-rating and specificity. When calculating these values, DP: true-positive, DN: true-negative, YP: falsepositive and YN: false-negative FN: false-negative is used. The true-positive expression refers to the number of operations in the sample space where the work carried out in the dissertation project, in which the model also contains smoke for visual data that actually contains smoke. The number of true-negative indicates the number of visual data that the model classifies as smoke-containing. On the one hand, the false-positive does not actually contain smoke; on the other hand, the false-negative characterizes the number of visual data classified by the model as though it actually contains smoke, though it does not contain smoke. The fact that the true-positive and true-negative values are high shows that the model produces realistic results. Models with a large number of false-positive numbers are likely to be triggered by the fire department in the absence of fire, and models with a large number of false-negative numbers that can lead to negligence by not seeing the fire. In this case, conducting such a numerical analysis before deciding on the model to be used is of great importance for the efficiency of the project. For all four models, total accuracy, positive interpretation power, negative interpretation power, sensitivity, F-evaluation and specificity values of the models were calculated based on the results of the estimation values table prepared using the values recorded in the output table. Total accuracy is the most intuitive and basic of the evaluation criteria. A number of analysis procedures were carried out on four models run on the same control group to choose between the different percentage of training and test data dropped and four different models trained. Estimation success of the models in different respects was measured and models were compared to each other and one of them was preferred to be used. The preferred model is the third model (M3), trained with 70% training and 30% test

data. Table 2 demonstrates the overall results of the corresponding physical experiments. According to the which, Model 3 is determined as the best model for all evaluation criterias. For the physical experiment M3 model was loaded on Raspberry Pi and test studies were carried out. In order to carry out the flight test in accordance with the rules, near the Kızılcahamam Soguksu Mili Park, a remote and sheltered area was preferred, and the test flight and related shots were carried out using daylight. As a smoke source, it was ignited with brazed gel in footed enamel brazier, and some plastic waste material was added for the brushing and bulking of smoke illustrates experimental result both validating the performance of the drone and its smoke recognition capacity. During the test flight of the system, images of smoke and vehicle were taken both from inside Raspberry and from outside. Since the vehicle does not have a vision system (FPV, First Person View) that can help the pilot to control the vehicle more easily, it has been very difficult to center the smoke in the In addition, due to vibration obtained images are blurry. However, even though the smoke recognition model worked flawlessly, negligible localization errors were observed and the similarity rate was not successful in each smoke input, the test flights was successfully completed.

Conclusions

This paper introduces a new low-cost drone equipped with image processing and object detection abilities for smoke and fire recognition tasks in forests. The statistics and results of the mathematical evaluations that the system is trained to give fast and low error rates are given in this section. The ssdlite mobilenet model, which is a sub-model of the supervised deep learning model called Model Zoo, which contains pre-educated models in the Tensorflow library, is taken as an example to be educated in the thesis project. There is a speed-precision trade-off that creates a kind of balance between sub-models. The criterion to be considered when selecting a model is the purpose for which the model will be used. Because in programming there is always a trade-off balance that makes it necessary to waive certain things while improving certain things. In the real-time object recognition application, the speed of the model is in the foreground and the accuracy rate remains in the minor preferences. In the project, the models that were trained with COCO training set were investigated and

the ssdlite mobile net model which was most likely to give the best results when working on real time smoke data on Raspberry was modified. When evaluating the suitability of the model used, it should be taken into consideration that each model cannot work with the same speed and accuracy rate in each data set. A model based on the dimension of colors cannot achieve the success of distinguishing the objects of color that distinguishes itself from the background on objects that are very close to the background. The preferred model has successfully performed its task. Based on these results, it was considered appropriate to use the drone and the early fire detection system for the detection of forest fires. In order to improve the system, it is possible to increase the maximum flight time of the unmanned aircraft so that the area that can be controlled in a single flight can be expanded. In this eco-friendly project, the system can be supplied from convertible energy sources such as solar energy instead of LiPo battery. In addition, in cases where smoke is detected in the unmanned aircraft, it is possible to estimate how far the smoke is from the vehicle by means of monocular depth detection algorithms, and this improvement enables the location of the fire to be reported to the ground station. In addition to the smoke recognition application, the same model can be trained to detect flame and fire visuals, and a downward camera can be added to the unmanned aerial vehicle to perform a downward fire / flame scan on the vehicle suspended above the smoke detection zone. This will enhance the accuracy of the project outcome. Hoping that the study will shed light on similar scientific studies.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Existing system

- Correlation based analysis and transformation methods
- Data Mining Approach to Predict Forest Fires using Meteorological Data SVM classifier

- Circle-based Approximation to Forest Fires with Distributed Wireless Sensor Networks using clustering
- Event Detection in Wireless Sensor Networks for Disaster Management

Draw back

Dense smoke tends to only obscure part of the light beam, whereas less-dense smoke particles can move around more freely, obscuring more of the light beam and therefore, impacting the alarm system.

3.2 PROPOSED SYSTEM:

- Here in fire detection, we are using the Haar Cascade classifier, which is very popular in object detection through the image or any other video feeds.
- Detection – open cv - python
- Haar-like Features
- Integral Image
- Training and Learning
- Adaptive Boosting (Ada Boost)
- Cascading

ADVANTAGES:

- The system can be used in various areas like in parking areas, road traffic congestions, agriculture, farming, there are many uses of object detection.
- In future, this approach can be extended by for other disasters as well.

3.3 SYSTEM REQUIREMENTS

3.3.1 HARDWARE REQUIREMENTS:

- System : Pentium i3 Processor.
- Hard Disk : 500 GB.

- Monitor : 15'' LED
- Input Devices : Keyboard, Mouse
- Ram : 2 GB

3.3.2 SOFTWARE REQUIREMENTS:

- Operating system: Windows 7.
- Coding Language: python
- Tool: Anaconda Navigator
- Libraries: OpenCV

CHAPTER 4

4.1 PROPOSED WORK

STEP 1: we capturing the video

STEP 2: read a new frame

STEP 3: video capturing

STEP 4: fire detection

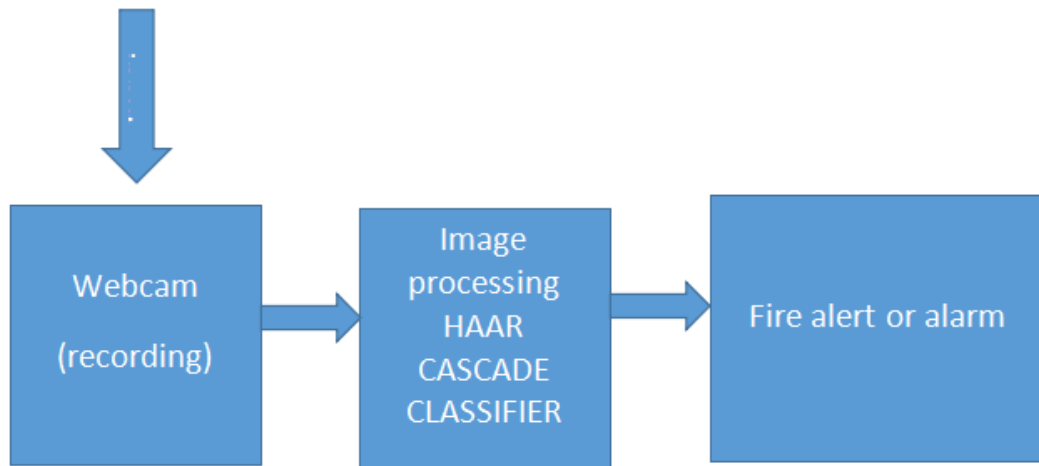
Step 5: fire detection

Step 6: mail sending

Step : alart

\

4.2 Architecture diagram



4.3 Algorithm and method

4.3.1 CNN(Convolutional Neural Network)

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The benefit of using CNNs is their ability to develop an internal representation of a two-dimensional image. This allows the model to learn position and scale in variant structures in the data, which is important when working with images. CNN is mainly used in image analysis tasks like Image recognition, Object detection & Segmentation.

IMPLEMENTATION

5.1 MODULUS

- DETECTION
- VIDEO STREAMING
- FIRE DETECTION
- MAIL SENDING & ALART

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

FIRE DETECTION

- the free encyclopedia. Fire detectors sense one or more of the products or phenomena resulting from fire, such as smoke, heat, infrared and/or ultraviolet light radiation, or gas. In dwellings, smoke detectors are often stand-alone devices.

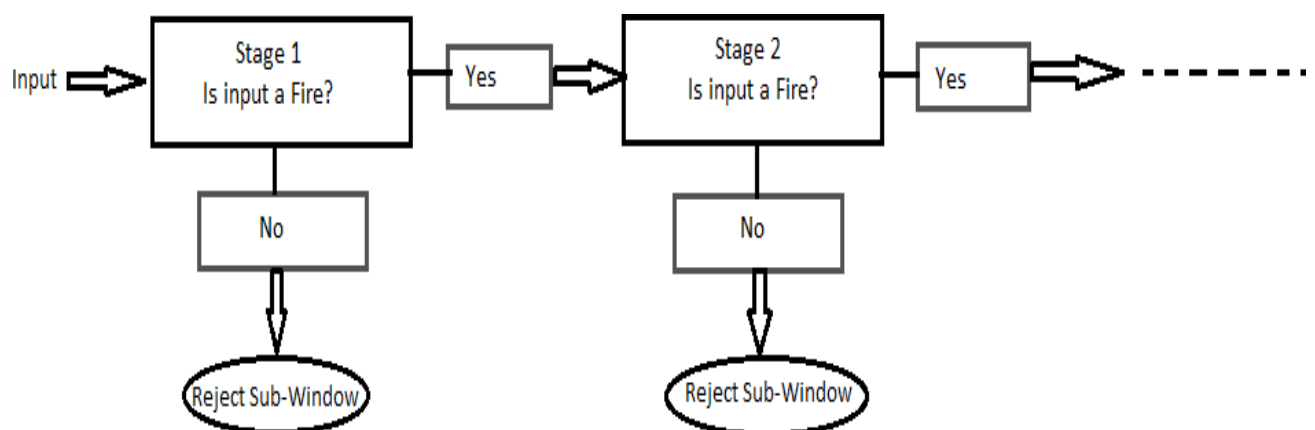
```
img_RGB = cv2.cvtColor(frame,cv2.COLOR_BGR2RGB)
img_cap = plt.imshow(img_RGB)
plt.show()
live_Camera.release()
cv2.destroyAllWindows()
```

You'll notice one thing that when hovering your cursor over the image, RGB values (i.e. shown in the red circles on image) changes.

matchstick) lies between those two values. (i.e. lower_region & upper_region)

Training and Learning:

- Extracting features from an image to detection. The machine is now trained to identify the features from the input. So it is fed with lots of information so that it can predict objects from the information.
- To train the machine, It needs a lot of pictures of fire, different fireplaces, conditions of fire, also known as a positive image. Concerning that, many negative images were uploaded.
- The negative image refers to pictures in which fire is not included in an image.
- Negative images are uploaded so it can differentiate between the two classes, which helps to an algorithm which features are more likely to be a fire and which are not expected to be fire.



alarm:

- `pygame.mixer.music.load("alarm.wav")`
- Importing all the libraries and modules required.
- 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.
- Create a display window for user input.

MAIL SENDING

Fire Mail is an easy to use npm module to send e-mail notifications based on entries in firebase.

5.2 RESULT AND DISCUSSION

5.2.1 RESULT

To evaluate the performance of the proposed model as a simulation tool, we utilized MATLAB and tested it using a diverse dataset containing videos in a wide variety of scenarios. The dataset includes videos containing the fire at different illumination levels and videos with ordinary objects that display fire-like color. The dataset was prepared predominantly using videos from the Mivia dataset with some videos from Zenodo, and the rest were taken from YouTube or recorded personally. The videos can be broadly divided into three categories: indoor, outdoor, and forest fires.

5.2.2 EXPERIMENTAL RESULT

The proposed model yields a comparatively greater true positive rate in scenarios where the fire is situated in a distant region. It excels over the competing models in these cases because these models employ a stricter set of color segmentation rules in their first phase of analysis. The restrictive color rules fail to extract the small fire regions in the scenarios where the fire is located far distant. Consequently, the comparatively liberal color rules employed in the proposed model yield a better true positive rate.

CHAPTER 6

SUMMARY AND CONCLUSION

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

In this section, we discuss the demonstration of a quantitative analysis to compare the performance of different strategies. We compared our method with well-known fire detection algorithms, which are based on YOLO networks and DL approaches. We used the results in their papers for comparison, but we are not sure whether they are true because source codes and datasets of these methods are not publicly available to check the real performance. We computed metrics such as F-measure (FM), precision, and recall, as in our earlier study [4]. 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 (3). 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 following equations can be used:

REFERENCES

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.

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.

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. IEEE/RSJ International Conference on. IEEE, 2019.

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.

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.

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.

P. V. K. Borges, J. Mayer, and E. Izquierdo, "Efficient visual fire detection applied for video retrieval," in *Proc. 16th Eur. Signal Process. Conf.*, Aug. 2008.

B. U. Toreyin and A. E. Cetin, "Online detection of fire in video," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2007, pp. 1–5.

C. Liu and N. Ahuja, "Vision-based fire detection," in *Proc. Int. Conf. Pattern Recognit.*, vol. 4. Aug. 2004, pp. 134–137.

T. Celik, H. Ozkaramanli, and H. Demirel, "Fire pixel classification using fuzzy logic and statistical color model," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, vol. 1. Apr. 2007, pp. 1205–1208.

T. Chen, P. Wu, and Y. Chio, "An early fire-detection method based on image processing," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3. Oct. 2004, pp. 1707–1710.

S. C. Liew, A. Lim, and L. K. Kwok, "A stochastic model for active fire detection using the thermal bands of modis data," *IEEE Geosci. Remote Sensing Lett.*, vol. 2, no. 3, pp. 337–341, Jul. 2005.

Y. Li, A. Vodacek, R. L. Kremens, A. Ononye, and C. Tang, "A hybrid contextual approach to wildland fire detection using multispectral imagery," *IEEE Trans. Geosci. Remote Sensing*, vol. 43, no. 9, pp. 2115–2126, Sep. 2005.

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.

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.

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.

J. Dozier, "A method for satellite identification of surface temperature fields of subpixel resolution," *Remote Sens. Environ.*, vol. 11, pp. 1–229, 1981.

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 fire monitoring from EOS-MODIS," *J. Geophys. Res.*, vol. 103, pp. 32 215–32 238, 1998.