International Journal of New Technology and Research (IJNTR)
ISSN: 2454-4116, Volume-7, Issue-4, April 2021 Pages 01-05

Early Forest Fire Detection Using Machine Learning Algorithms

Pradeep Kumar G, Rahul R, Ravindharan N

***Abstract*— Apart from inflicting tragic loss of lives and valuable natural and individual properties as well as thousands of hectares of forest and many homes forest fires are an excellent menace to ecologically healthy big forests and to the atmosphere. Every year, thousands of fire across the world cause disasters on the far side live and outline. This issue has been the analysis interest for several years, there are a large quantity of good studied solutions accessible out there for testing or maybe prepared to be used to resolve this downside. Forest and concrete fires are major problem for several countries within the world. Currently, there are many different solutions to detect the forest fires. People are using sensors to detect the fire. But this case is not possible for large acres of forest. In this paper, we discuss a new approach for fire detection, in which modern technologies are used. In particular, we propose a platform that is the Artificial Intelligence. The computer vision methods for recognition and detection of smoke and fire, based on the still images or the video input from the cameras. Machine learning for finding the output. The accuracy is based on the algorithms which we are going to use and the datasets and splitting them into train set and test set.**

***Index Terms*—Accuracy, KNN Algorithm, Random Forest Algorithm, Segmentation.**

I. INTRODUCTION

Forest area covers are the protectors of earth's ecological balance. Sadly, the fire is determined typically once it's already contact an outsized space, creating its management and stoppage arduous and even not possible now and then. The result's devastating loss and irreparable harm to the surroundings and atmosphere (30% of carbon dioxide (CO2) within the atmosphere comes from forest fires), additionally to irreparable harm to the ecology (huge amounts of smoke and carbon dioxide (CO2) within the atmosphere). Among different terrible consequences of forest fires unit of measurement semi-permanent fateful effects like impacts on native weather patterns, heating, and extinction of rare species of the flora and fauna. Fast and effective detection may well be a key for firefighting. The 2019–2020 Australian bushfire season, a colloquially said as Black Summer, was an intense bushfire in many parts of Australia. Throughout the summer, several fires burnt, primarily at intervals the southeast of the country. the key fires peaked throughout December–January. The fires burnt approximately eighteen million hectares destroyed over six thousand buildings and killed a minimum of 34 people. Nearly three billion terrestrial vertebrates alone – the overwhelming majority being reptiles – were affected and many species were believed to be driven into extinction.

Ecologists from The University of Sydney calculated nearly 480 million mammals, birds, and reptiles were lost since Sept with issues that entire species of plants and animals may be dead in the bushfire, later enlarged to over a billion. National Aeronautics and Space Administration calculated that the quantity of dead koalas may be as high as twenty-five thousand on the island, one-fourth of the beehives of the Liguria region honey bees that settled the Island were believed to have been destroyed. In February 2020 it had been estimated that researchers from Charles Sturt University found that the deaths of nine smoky mice were from severe respiratory illness caused by smoke haze that contained PM2.5 particles (equivalent of 37 cigarettes) coming from bushfires fifty kilometers away.

At its peak, air quality fallen to unsafe levels in south-east states. Nearly eighty % of Australians were affected either directly or indirectly by the bushfires. By January 2020, the smoke had affected eleven thousand kilometers across the Pacific to Chile and Argentina. National Aeronautics and Space Administration calculated that 306 million of carbon dioxide gas had been emitted. Throughout the subsequent crisis, associate air tanker and a couple of helicopters crashed throughout firefighting operations, the air tanker crash resulting in the deaths of the three crew. A pair of fire trucks were caught in fatal incidents caused directly by forest fire conditions, killing three people. An approximate amount of $500 million was given by the general public, international organizations, public figures and celebrities for victim relief and life recovery. Convoys of given food, dresses and livestock feed were sent to the affected areas. Thus to avoid like this uncontrollable wide spreading of forest fire situations it's necessary to find fires in associate early state and to forestall the propagation. Therefore, the surroundings and the atmosphere could be saved from fatal disaster.

Thus we've used a technique for the detection of fire and smoke from the given video by the method of image processing techniques and machine learning classification algorithms. The aim of this project is to style a technique that is less complex for the detection of fire and smoke from the forest affected video with high accuracy. Initially, KNN algorithm was accustomed to predict the fire and smoke and then Random Forest algorithm was accustomed improve the accuracy.

Pradeep Kumar G, Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, India
Rahul. R, Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, India
Ravindharan N, Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, India

II. METHODS

A. KNN Algorithm

K-Nearest Neighbor (KNN) could be a simplest algorithm that stores all on the market cases and classifies the new information or the cases supported a similarity case. KNN is employed in search application wherever we tend in searching for similar things (i.e.,) If the task is in some sort of fine things almost like this one then we tend to decision it as a KNN search. K denotes the amount of nearest neighbor of testing information. e.g., Let us take 2 categories as category A and category B. There is a new data point and that we have to be compelled to predict whether it belongs to category A or category B. Suppose if K=3 then the 3 nearest category to the new data point is checked and therefore the most typical category is allotted to the testing data point. The nearest category is found by scheming the Euclidian distance. A number of the applications of KNN in day to day life is that the use of KNN in recommendation system like Amazon, Netflix. Over thirty-five percent of Amazon revenue is from the recommendation. it's additionally employed in idea search or looking symmetrically documents containing similar topics. There are a unit billions of data information increasing day by day. Every document on the web has multiple ideas that would be potential idea. Currently to extract the idea from a group of documents KNN is employed. KNN could be a lazy learner as a result of it doesn't have discriminative perform from the training information. It memorizes the training information and there's no learning section of the model. The work is happened at the time of request.

B. Random Forest Algorithm

Random Forest is a method that operates by constructing multiple decision trees during training phase. The decision of the majority of trees is chosen by the random forest at the final decision. Decision tree is a tree shaped diagram used to determine a course of the action. Each branch of the tree represents a possible decision occurrence or action. e.g., Suppose there are 3 fruits in a bowl 1 such as cherry, orange, apple. We need to separate it. So first we are using condition “diameter>=3” so cherry is separated because the cherry diameter is small. Then orange and apple is mixed. Then using another condition “Is color orange?”. So if true orange is separated and false as apple. Finally, we get orange. Similarly, if we get apple in bowl 2 and again orange in bowl 3. Here the majority is orange among the 3 bowls and hence orange is the majority and it is the final decision. In random forest use of multiple trees reduce the risk of overfitting. Training time is less. It has high accuracy which runs efficiently on large database. For large data it produces highly accurate predictions. It can maintain accuracy when large proportion of data is missing.

C. Segmentation

In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple segments. The goal of segmentation is to simplify and to change the representation of an image into something that is more meaningful and easier to analyze. It assigns a label to every pixel in an image such that pixels with the same label share certain characteristics. Image Segmentation could be used for object recognition, face recognition, medical image analysis, machine vision applications etc. There are various types of segmentation methods such as threshold method, edge based segmentation, clustering methods, graph based methods etc. OpenCV usually captures images and videos in 8-bit, unsigned integer, RGB format. In alternative words, captured pictures are often thought of as three matrices, Red, Green and Blue with whole number values ranges from zero to Two hundred and fifty-five. As an example, RGB (0, 0, 255) is rendered as blue, as a result of the blue parameter is about to its highest price (255) and therefore the other units are set to zero. RGB image is made within the image, every tiny box represents a picture element of the image. In real pictures, these pixels’ area unit is so tiny such that human eye cannot differentiate. So HSV color area is employed that consists of three matrices, 'hue', 'saturation' and 'value'. In OpenCV, price vary for 'hue', 'saturation' and 'value' area unit severally 0-179, 0-255 and 0-255. 'Hue' represents the color, 'saturation' represents the number to that that individual color is mixed with white and 'value' represents the number to that that individual color is mixed with black. Hue part makes the algorithmic rule less sensitive (if not invariant) to lighting variations. It's a tool freelance color illustration format. The HSV color illustration is beneficial to find specific color varieties, e.g., skin coloring, fireplace color etc.

III. IMPLEMENTATION

A. Dataset Preparation

The datasets are taken from the Kaggle which supports a variety of dataset publication formats where the dataset publishers share their data in an open, accessible format. The dataset taken is in the format of CSV file. These are the most common of the file formats available on Kaggle and are the best choice for tabular data. CSV files will have tabular descriptions of rows and columns. Here the dataset file consists of three columns where the first two columns consists of namely fire and smoke which are the independent variables and the fire status which is the dependent variable. The fire status depends on the value of fire and smoke and so it is the dependent variable. Here dataset file consists of nearly thousand rows of data for the corresponding three columns. Also it consists of videos for testing to find the output of the fire status. From the datasets obtained, the dependent variable is passed into the label encoder.

B. Label Encoder

It is a part of the SciKit Learn library in Python, and are used to convert categorical data or text data into numbers, where our predictive models can be understood better. The third column will be the fire status column (y), which is in the text format. The text or string values in our data if we’re going to run any kind of model on it error will mostly happen. So before running the model, there is a need to make this data ready for the model. To convert the text or string data into numerical data, we use the Label Encoder class. To label encode the last column, import the LabelEncoder class from the sklearn library, fit, transform the last column of the data and then replace the existing text data with the new encoded data. Hence the fire status which is in the text form is converted into the numerical data as the “No fire and No smoke” fire status is taken as the value 2. Similarly, “Smoke” as the value 3, “Fire and Heavy Smoke” is taken as the value 0

and the "Heavy Fire and Heavy Smoke" is taken as the value 1.

C. Train Test Split

Then the dependent variable from the label encoder which is converted into numerical data and the independent variable both together is split into training and testing data. The training data is implemented to build up a model, while the testing data is used to validate the model built. The 75% of the data is taken as the training data randomly and the remaining 25% of the data is taken as the testing data randomly. The random state is used to split the dataset randomly. You can change the split percentage as per choice, but it is advisable to give at least 60% data as train data for good results. X train and Y train are train datasets which is the 75% of the training data. X test and Y test are test datasets which is the 25% of the testing data.

D. KNN Algorithm Implementation

In Python necessary libraries are imported for the KNN algorithm. Then numpy libraries for used for arrays and matrices. For reading the dataset file pandas library is used. Then importing 2 machine learning libraries KNeighborsClassifier from sklearn.neighbors to implement the K nearest neighbor and accuracy from sklearn.metrics for accuracy classification score. The value of K is randomly taken as 3. There's no specific procedure to calculate the K value, thus we'd like to undertake some values to search out the simplest out of them. A really low K value may lead to results of outliers within the model and noisy. A bigger value for K might be good, however it's going to face some difficulties.

E. Random Forest Algorithm Implementation

In Python necessary libraries are imported for the Random Forest algorithm. Then numpy libraries for used for arrays and matrices. For reading the dataset file pandas library is used. Then importing 2 machine learning libraries RandomForestClassifier from sklearn.ensemble to algorithm and accuracy for sklearn.metrics for accuracy classification score. n_estimators = 10, the required number of trees is taken in the Random Forest. value 10 is taken as default. Any number can be chosen but the overfitting issue should be checked.

F. Confusion Matrix

A confusion matrix is a visualization of prediction results on a classification problem whether the algorithm has predicted correctly or not. The count values are used to summarize the correct and incorrect predicted values. The confusion matrix visualizes in which your classification model is confused when it makes predictions. Four cases are there and as follows.

- True Positives (TP): These are cases in which we predicted yes (there will be fire and smoke) and they do have fire and smoke.
- True Negatives (TN): We predicted no, and there will be no fire and smoke.
- False Positives (FP): We predicted yes, but there will be no fire and smoke.
- False Negatives (FN): We predicted no, but there will be fire and smoke.

The testing accuracy for the algorithm is calculated from the below formula:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

G. Accuracy

Hence 75% of the datasets are used to train the model of both the algorithm. Hence a KNN algorithm model and Random Forest algorithm model is built with X train and Y train. So the training accuracy for both the algorithms is predicted. To test the model 25% of the datasets are used where X test is passed to the trained model and so Y predict will be get as an answer for the corresponding X test. Now the Y test and Y predict will be compared and so the testing accuracy will be predicted for both the algorithms. Confusion matrix is used to predict the testing accuracy for both the algorithms.

H. Segmentation

A video file downloaded from the Kaggle is played. Then the video is read and there will be two variables grabbed and img. Grabbed has no value and it is omitted and video is stored in img. Then the video is cut into frame by frame and it is resized. Once again in img the resized image is stored. Usually, one will assume that RGB colors are appropriate for color primarily based segmentation. However, HSV colors are the most fitted color for color based image segmentation. Hence from the image (frame by frame) of the video RGB values are converted to HSV values. All are co-related to the color luminance (what we call intensity) in R, G, B of RGB colors, (i.e.,) luminance cannot be separated from the color information. To separate image luminance from color information HSV is employed. Working on or there is a need for luminance of the image/frame HSV makes it easier and simple. HSV additionally employed in things wherever color description plays a major integral role. Hence frame by frame, the RGB values are converted to HSV values. We threshold the HSV image for a range of fire and smoke color. Then to mask the fire in the image which will be getting from frame by frame extract the red, yellow and the combination of both from the given range using the hsv range color of red and yellow. The red region will be masked based on the range of the value and then the number of non-zero pixels is counted from the red region of the image and it is stored in red. Similarly, for yellow region and the values will be stored in yellow. Finally, the combination of red and yellow will be masked and the overall pixels of fire is calculated and stored in fire. Similarly, to mask the smoke in the image that will be getting from frame by frame first the smoke is masked using the given range and the number of non-zero pixels is counted and stored in smk. Then for lsmk (light smoke) and bsmk (black smoke). Then the combination of lsmk and bsmk is stored in mask7. Again the combination of smk and mask 7 is stored in mask 8. So in mask 8 all the smoke varieties will be there and the non-zero pixels is counted and its value is stored in smokee. Finally, the smoke and fire values from masking is given to the classifier algorithm and both the classifier algorithm gives the output for the given values. Among the four fire status one will be printed as the output frame by frame from the video. Finally, cv2.imwrite is used to store the videos played and it can be seen in "target.png". cv2.imshow is used to play the video externally otherwise it will be played

internally and cannot be seen. When the video is playing and to stop the video when "q" is pressed the loop will be halted and the video will be stopped.

IV. Results And Discussions

A. Dataset Preparation

The dataset was obtained from the Kaggle. The datasets consist of 997 rows of data where 747 rows of data are used for training the model and 250 rows of data are used for testing the trained model.

B. Label Encoder and Train Test Split

In Label Encoder Preprocessing the four different fire status is converted into binary numerical value and it is represented like in a matrix for all the data's. For both algorithms the training accuracy is found from X train and Y train. Then the Y test and Y predict is compared. Based on it the confusion matrix is developed for the visualization of the correct and incorrect predictions of the algorithm and then finally the testing accuracy is found for the algorithms.

C. KNN Algorithm Confusion Matrix and Accuracy

```
 K NEAREST NEIGHBOUR CLASSIFICATION
TRAINING ACCURACY
K Neighbors Classifier Training Accuracy: 0.9906291834002677
Y_test
[3 1 2 3 0 1 0 2 1 3 0 3 2 2 1 1 0 0 0 2 1 2 3 2 2 1 0 3 1 0 1 1 2 1 0 0 1
 0 3 1 3 1 3 3 2 0 1 1 1 0 0 1 0 0 1 3 1 3 3 0 3 1 3 3 1 3 2 3 3 2 0 3 3 1
 3 0 3 3 0 0 3 0 1 3 3 0 1 3 3 3 0 2 1 1 0 1 1 0 2 3 0 1 3 3 2 3 1 1 2 0 2
 1 3 0 0 3 3 0 3 3 1 3 3 1 1 1 3 3 3 3 1 2 0 1 2 3 1 3 2 3 1 3 2 2 1 2 3 1
 3 0 0 1 3 0 1 0 1 0 3 1 1 0 0 3 1 2 1 0 1 3 3 0 2 1 1 0 3 1 3 1 3 3 0 3 1
 1 2 0 1 1 0 0 2 3 3 1 0 3 3 3 0 0 1 2 1 2 0 1 1 2 2 2 3 3 1 0 3 1 0 0 2 3
 1 0 1 1 2 2 0 2 2 0 3 2 0 1 0 3 2 1 3 1 0 2 3 2 3 3 3 3 0]
Predicted output of X_test
[3 1 2 3 0 1 0 2 1 3 0 3 2 2 1 1 3 0 0 2 1 2 3 2 2 1 0 3 1 0 1 1 2 1 0 0 1
 0 3 1 3 1 3 3 2 0 1 1 1 0 0 1 0 0 1 3 1 3 3 0 3 1 3 3 3 3 2 3 3 2 0 3 3 1
 3 0 3 3 0 0 3 0 1 3 3 0 1 3 3 3 3 2 1 1 0 1 1 0 2 3 0 1 3 3 2 3 1 1 2 0 2
 1 3 0 0 3 3 0 3 3 1 3 3 1 1 1 3 3 3 3 1 2 0 1 2 3 1 3 2 3 1 3 2 2 1 2 3 1
 3 3 0 1 3 0 1 0 1 0 3 1 1 0 0 3 1 2 1 0 1 3 3 0 2 1 1 0 3 1 3 1 3 3 0 3 1
 1 2 0 1 1 0 0 2 3 3 1 0 3 3 3 0 0 1 2 1 2 0 1 1 2 2 2 3 3 1 0 3 1 0 0 2 3
 1 0 1 1 2 2 0 2 2 0 3 2 0 1 0 3 2 1 3 1 0 2 3 2 3 3 3 3 0]
CONFUSION MATRIX
[[57  0  0  3]
 [ 0 71  0  1]
 [ 0  0 41  0]
 [ 0  0  0 77]]
TESTING ACCURACY
Accuracy: 0.984
```

Fig. 1.KNN algorithm confusion matrix and accuracy

D. Random Forest Confusion Matrix and Accuracy

```
 RANDOM FOREST CLASSIFICATION
TRAINING ACCURACY
Random Forest Classifier Training Accuracy: 0.998661311914324
Y_test
[3 1 2 3 0 1 0 2 1 3 0 3 2 2 1 1 0 0 0 2 1 2 3 2 2 1 0 3 1 0 1 1 2 1 0 0 1
 0 3 1 3 1 3 3 2 0 1 1 1 0 0 1 0 0 1 3 1 3 3 0 3 1 3 3 1 3 2 3 3 2 0 3 3 1
 3 0 3 3 0 0 3 0 1 3 3 0 1 3 3 3 0 2 1 1 0 1 1 0 2 3 0 1 3 3 2 3 1 1 2 0 2
 1 3 0 0 3 3 0 3 3 1 3 3 1 1 1 3 3 3 3 1 2 0 1 2 3 1 3 2 3 1 3 2 2 1 2 3 1
 3 0 0 1 3 0 1 0 1 0 3 1 1 0 0 3 1 2 1 0 1 3 3 0 2 1 1 0 3 1 3 1 3 3 0 3 1
 1 2 0 1 1 0 0 2 3 3 1 0 3 3 3 0 0 1 2 1 2 0 1 1 2 2 2 3 3 1 0 3 1 0 0 2 3
 1 0 1 1 2 2 0 2 2 0 3 2 0 1 0 3 2 1 3 1 0 2 3 2 3 3 3 3 0]
Predicted output of X_test
[3 1 2 3 0 1 0 2 1 3 0 3 2 2 1 1 0 0 0 2 1 2 3 2 2 1 0 3 1 0 1 1 2 1 0 0 1
 0 3 1 3 1 3 3 2 0 1 1 1 0 0 1 0 0 1 3 1 3 3 0 3 1 3 3 1 3 2 3 3 2 0 3 3 1
 3 0 3 3 0 0 3 0 1 3 3 0 1 3 3 3 0 2 1 1 0 1 1 0 2 3 0 1 3 3 2 3 1 1 2 0 2
 1 3 0 0 3 3 0 3 3 0 3 3 1 1 1 3 3 3 3 1 2 0 1 2 3 1 3 2 3 1 3 2 2 1 2 3 1
 3 0 0 1 3 0 1 0 1 0 3 1 1 0 0 3 1 2 1 0 1 3 3 0 2 1 1 0 3 1 3 1 3 3 0 3 1
 1 2 0 1 1 0 0 2 3 3 1 0 3 3 3 0 0 1 2 1 2 0 1 1 2 2 2 3 3 1 0 3 1 0 0 2 3
 1 0 1 1 2 2 0 2 2 0 3 2 0 1 0 3 2 1 3 1 0 2 3 2 3 3 3 3 0]
CONFUSION MATRIX
[[60  0  0  0]
 [ 1 71  0  0]
 [ 0  0 41  0]
 [ 0  0  0 77]]
TESTING ACCURACY
Accuracy: 0.996
```

Fig. 2. Random Forest algorithm confusion matrix and accuracy

Form the above Fig. 1 and Fig. 2 the values in the confusion matrix where the diagonals are the one which has the actual fire status and the predicted fire status same. They are the correctly predicted values. Other values in the matrix lead to the incorrect predictions because the actual fire status and the predicted fire status is not same.

The accuracy in the Fig. 2 is calculated as follows:

= Total no. of correct predictions / Total no. of predictions

= 249 / 250

= 0.996 (Testing Accuracy of Random Forest)

Similarly, testing accuracy is calculated from the KNN algorithm as above.

E. Outputs for the input video

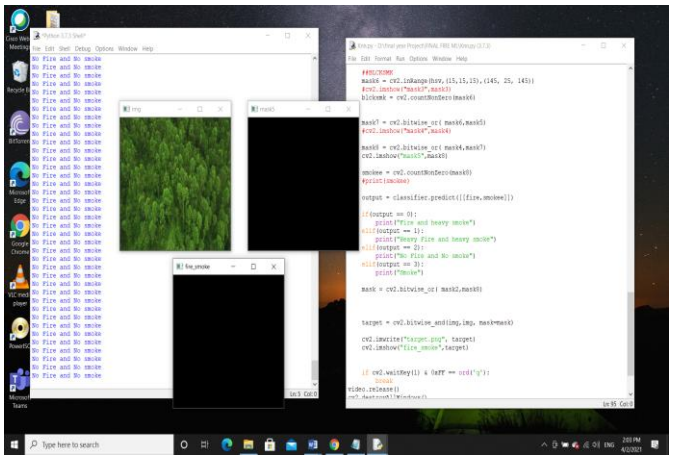

Fig. 3(a)

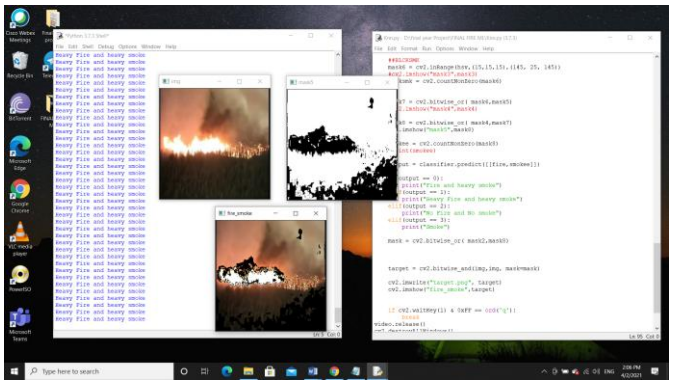

Fig. 3(b)

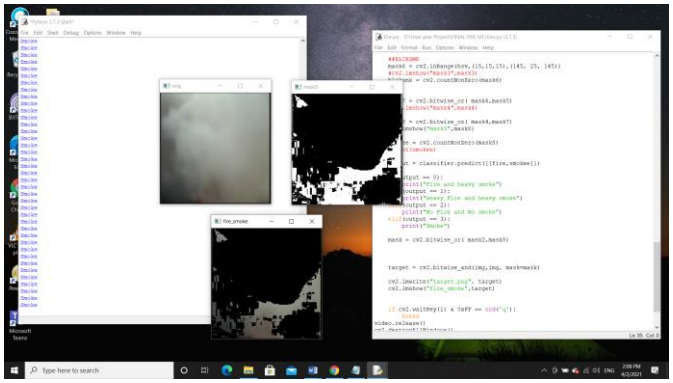

Fig. 3(c)

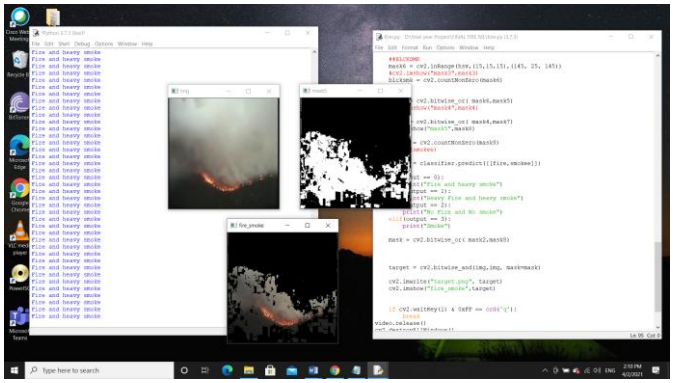

Fig. 3(d)

Fig. 3 Outputs for (a) No Fire and No Smoke (b) Heavy Fire and Heavy Smoke (c) Smoke (d) Fire and Heavy Smoke

Hence from the above Fig. 3 using masking technique in image processing and machine learning algorithms fire and smoke is detected and printed on the output console frame by frame.

Table 1. Accuracy of Different Classification Algorithms.

S.NO	ALGORITHM	ACCURACY
1.	KNN	98.4%
2.	Random Forest	99.6%

First KNN algorithm is implemented and it gives 98.4% accuracy and then to improve the accuracy better Random Forest algorithm is used which provides 99.6%. Hence the accuracy is improved.

.

V. CONCLUSION

In this study the two classification algorithms were used where the first one is the KNN algorithm which is implemented which gave the accuracy of 98.4% and then to improve the accuracy another algorithm is implemented (i.e.,) Random Forest algorithm which provides the accuracy of 99.6%. Hence the accuracy has increased to nearly 1.2%. Then image processing techniques such as segmentation is used where masking is implemented to predict the fire and smoke in the image frame by frame when a video is played. The output is displayed based on the four fire status continuously frame by frame. Future works can be carried out by sending an alert to the Forest department officers through mail by providing the exact location of fire and smoke if detected in the forest area to prevent the vast environmental effects and imbalance in the ecosystem.

VI. REFERENCES

[1] Sinha, Ditipriya, Kumari, Rina; Tripathi, Sudhakar, “Semisupervised Classification Based Clustering Approach in WSN for Forest Fire Detection.”, Wireless Personal Communications, Dec2019, Vol. 109 Issue 4, p2561-2605. 45p.

[2] Shivam Pareek, Shreya Shrivastava, Sonal Jhala, Juned A. Siddiqui, Savitanandan Patidar "IOT and Image Processing based Forest Monitoring and Counteracting System”, 4th International Conference on Trends in Electronics and Informatics (ICOEI), 2020.

[3] Tao Wang, Jianshuo Hu, Tingyu Ma, Jing Song "Forest fire detection system based on Fuzzy Kalman filter", International Conference on Urban Engineering and Management Science (ICUEMS), 2020.

[4] Sarah Masoumi, Thomas C. Baum, Amir Ebrahimi, Wayne S. T. Rowe, Kamran Ghorbani, "Reflection Measurement of Fire Over Microwave Band: A Promising Active Method for Forest Fire Detection”, IEEE Sensors Journal (Volume: 21, Issue: 3, Feb.1, 1 2021).

[5] Jianmei Zhang, Hongqing Zhu, Pengyu Wang, Xiaofeng Ling, "ATT Squeeze U-Net: A Lightweight Network for Forest Fire Detection and Recognition", IEEE Access (Volume: 9), January 2021.

[6] Alina-Elena Marcu, George Suciu, Elena Olteanu, Delia Miu, Alexandru Drosu, Ioana Marcu, "IOT System for Forest Monitoring", IEEE, 42nd International Conference on Telecommunications and Signal Processing (TSP), 2019.

[7] Li Jie, Xiao Jiang, "Forest fire detection based on video multi-feature fusion", IEEE, 2nd IEEE International Conference on Computer Science and Information Technology, 2009.

[8] Jing Wang, Weiguo Song, Wei Wang, Yongming Zhang, Shixing Liu, "A new algorithm for forest fire smoke detection based on MODIS data in Heilongjiang province", IEEE, International Conference on Remote Sensing, Environment and Transportation Engineering, 2011.

.