EMERGING METHODS FOR EARLY DETECTION FOR FOREST FIRES

IBM NALAIYATHIRAN(HX8001)

PROJECT REPORT

Submitted by

AISHWARYA E	**211719104008**
CHARU LATHA S K	**211719104029**
CHARULATHA T	**211719104028**
BRINDHA S	**211719104026**

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING IN
COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI INSTITUTE OF TECHNOLOGY
ANNA UNIVERSITY:
CHENNAI 600 025
NOVEMBER 2022

BONAFIDE CERTIFICATE

Certified that this project report **"EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRES"** , is the bonafide work of **AISHWARYA E(211719104008), CHARU LATHA S K(211719104029), CHARULATHA T (211719104028)** and **BRINDHA S(211719104026)** who carried out the mini project work under my supervision.

SIGNATURE:
Dr. T.RAJENDRAN
HEAD OF DEPARTMENT,
Associate Professor,
Dept. of Computer Science and Engg.,
Rajalakshmi Institute of Technology,
Kuthambakam post,
Chennai-600124.

SIGNATURE:
MS.N.KANAGAVALLI,
B.E., M.E.,
MENTOR,
Assistant Professor,
Dept. of Computer Science and Engg.,
Rajalakshmi Institute of Technology,
Kuthambakam post,
Chennai-600124

ACKNOWLEDGEMENT

We express our sincere gratitude to our honorable Chairperson **Dr.(Mrs.)THANGAM MEGANATHAN, M.A., M.Phil., Ph.D.,** and Chairman **Thiru. S.MEGANATHAN, B.E., F.I.E.,** for their constant encouragement to do this project and also during the entire course period.

We thank our Principal **Dr. P. K. NAGARAJAN** and our Head of the department **Dr. T.RAJENDRAN** for their valuable suggestions and guidance for the development and completion of this mini project.

Words fail to express our gratitude to our industry Mentor **MS.SHANTHI** and faculty Mentor **MS.N.KANAGAVALLI M.E.,** who took interest in our **IBM NALAIYA THIRAN** project and gave their consistent support and guidance during all stages of this project.

Finally, we thank all the teaching and non-teaching faculty members of the DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING of our college who helped us to complete this project.

Above all we thank our parents and family members for their constant support and encouragement for completing this project.

INDEX

INTRODUCTION

Forest fires are a major environmental issue, creating economic and ecological damage while endangering human lives. There are typically about 100,000 wildfires in the United States every year. Over 9 million acres of land have been destroyed due to treacherous wildfires. It is difficult to predict and detect Forest Fire in a sparsely populated forest area and it is more difficult if the prediction is done using ground-based methods like Camera or Video-Based approach. Satellites can be an important source of data prior to and also during the Fire due to its reliability and efficiency. The various real-time forest fire detection and prediction approaches, with the goal of informing the local fire authorities.

A system for automatically detecting fires in select areas, and reacting thereto to put out the fires. A stationary, earth orbit satellite, pilotless drone aircrafts or piloted aircraft contains one or more infrared detectors and optical means for detecting small fires when they first occur in fields and wooded areas, preferably where man made campfires and trash dumping are prohibited

LITERATURE SURVEY

Surapong Surit, Watchara Chatwiriya proposed a method to detect fire by smoke detection in video. This approach is based on digital image processing approach with static and dynamic characteristic analysis. The proposed method is composed of following steps :

1. The first is to detect the area of change in the current input frame in comparison with the background image.

2. The second step is to locate regions of interest (ROIs) by component algorithm, the area of ROI is calculated by convex hull algorithm and segments the area of change from image.

3. The third step is to calculate static and dynamic characteristics, using this result we decide whether the object detected is the smoke not. The result shows that this method accurately detects fire smoke.

Osman Gunay and Habiboglu proposed a system based on the Covariance Descriptors, Color Models, and SVM Classifier. This system uses video data. Spatio-temporal Covariance Matrix (2011) is used in this system which divides the video data into temporal blocks and computes covariance features.

Dimitropoulos (2015) proposed an algorithm where a computer vision approach for fire-flame detection is used to detect fire at an early stage.

Initially, background subtraction and color analysis is used to define candidate fire regions in a frame and this approach is a nonparametric model.

Following this, the fire behavior is modeled by employing various Spatio-temporal features such as color probability, flickering, spatial and spatiotemporal energy.

After flame modeling the dynamic texture analysis is applied in each candidate region using Linear Dynamical Systems, Histogram and Mediods.

LDS is used to increase the robustness of the algorithm by analyzing temporal evolution of pixel intensities. Pre-processing is done after this to filter non-candidate regions.

Spatio-temporal analysis is done to increase the reliability of the algorithm. The consistency of each candidate fire region is estimated to determine the existence of fire in neighboring blocks from the current and previous video frames.

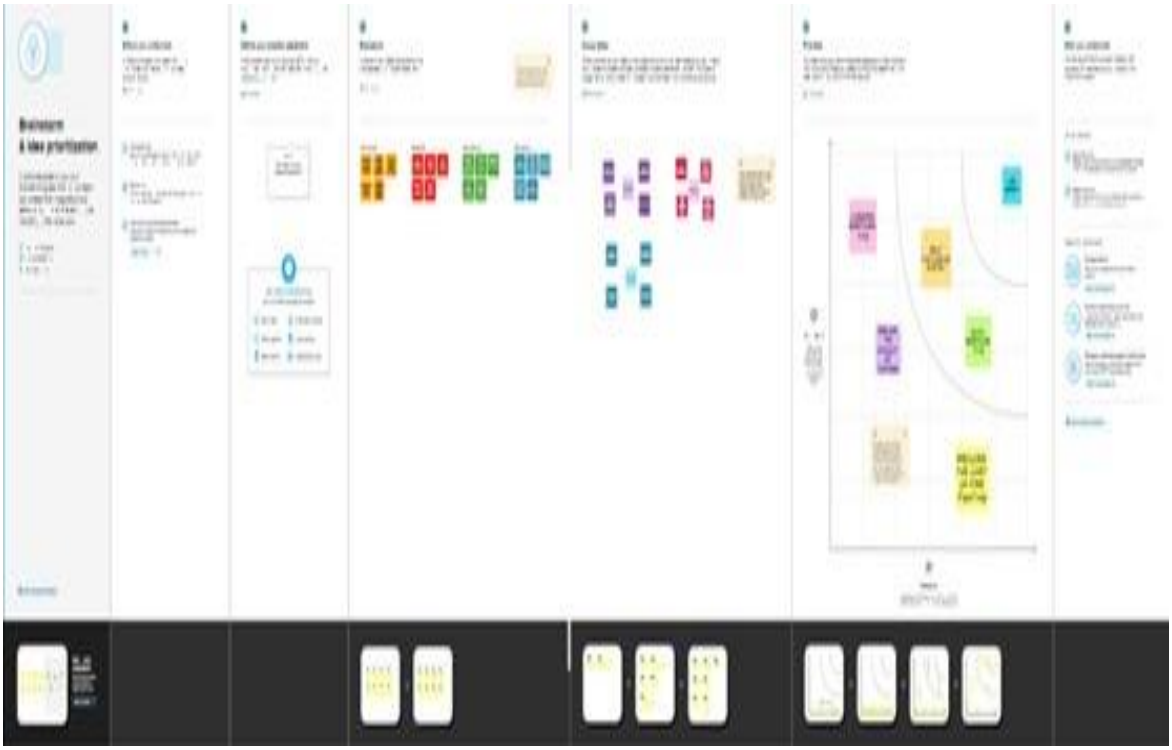
Celik (2007) proposed a generic model for fire and smoke detection without the use of sensors . Fuzzy based approach is used in this system. Color models such as YCbCr, HSV are used for fire and smoke detection. The fire is detected using YCbCr color model samples because it distinguishes luminance and chrominance. Y, Cb, Cr color channels are separated from RGB input image.

IDEATION AND PROPOSED SOLUTION

1. EMPATHY MAP CANVAS

An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user person, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.

2. IDEATION AND BRAINSTORMING



Ideation essentially refers to the whole creative process of coming up with and communicating new ideas. Ideation is innovative thinking, typically aimed at solving a problem or providing a more efficient means of doing or accomplishing something. It encompasses thinking up new ideas, developing existing ideas, and figuring out means or methods for putting new ideas into practice. Ideation is similar to a practice known as brainstorming

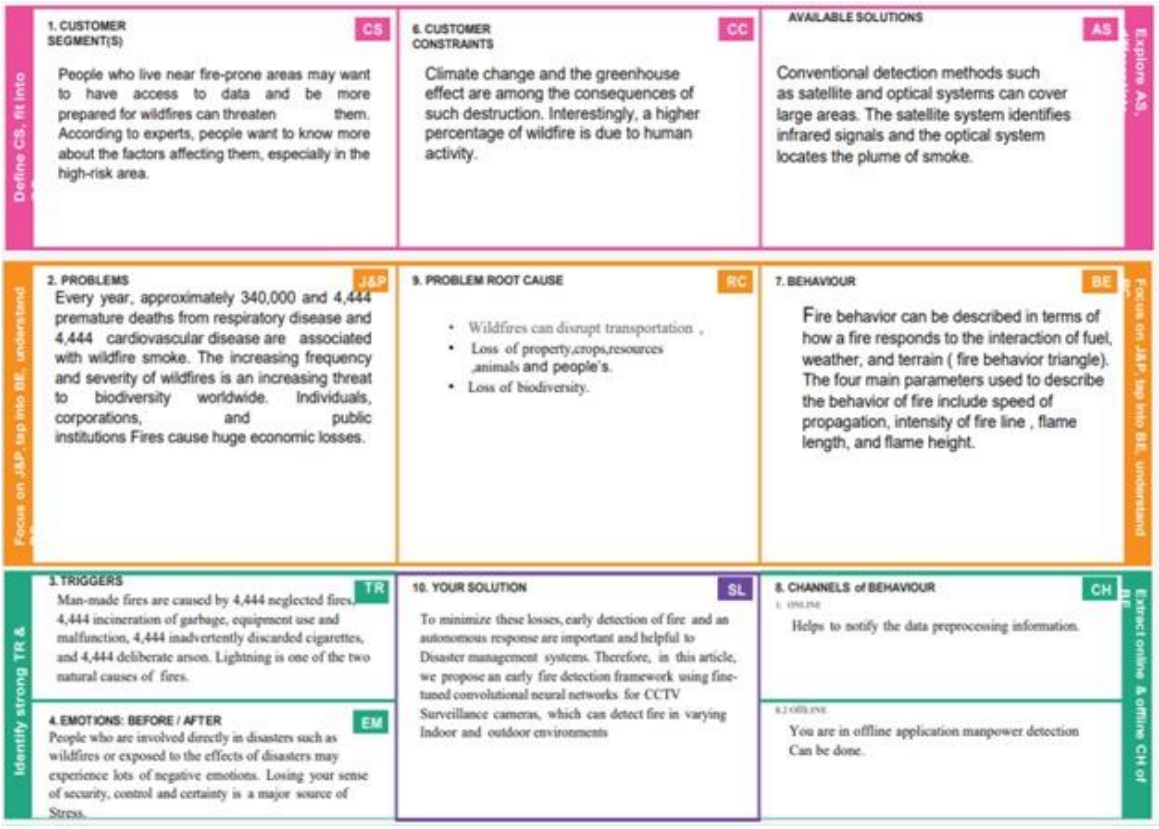
PROPOSED SOLUTION

S.NO	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	A forest fire risk prediction algorithm, based on support vector machines, is presented. The algorithm depends on previous weather conditions in order to predict the fire hazard level of a day.
2.	Idea / Solution description	Use computer vision methods for recognition and detection of smoke or fire, based on the still images or the video input from the drone cameras
3.	Novelty / Uniqueness	Real time computer program detect forest fire in earliest before it spread to larger area.
4.	Impact on society	Blocked roads and railway lines, electricity, mobile and land telephone lines cut, destruction of homes and industries.
5.	Business Model (Revenue Model)	The proposed method was implemented using the Python programming language on a Core i3 or greater (CPU and 4GB RAM.)

PROBLEM SOLUTION FIT

In this problem solution fit consists of the following segments:

1.Customer segment
2.Problems
3.Triggers
4.Emotions: Before/after
5.Available solutions
6.Customer constraints
7.Behaviour
8.Channels of behaviour
9.Problem root cause
10.Solution



REQUIREMENT ANALYSIS

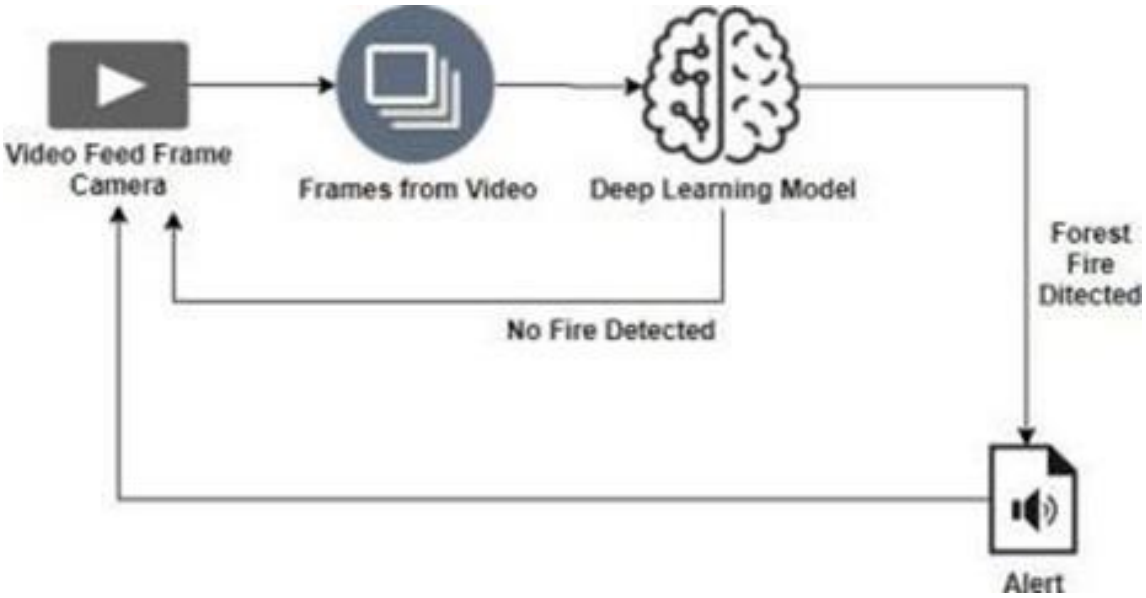
Functional requirements:

S.NO	FUNCTIONAL REQUIREMENTS	SUB REQUIREMENTS
1.	User Registration	Registration through Form Registration through Gmail
2.	User Confirmation	Confirmation via E-mail Confirmation via OTP
3.	Image Recognition	The system shall be able to take real inputs of satellites images and determine whether image contains fire or not.
4.	Forest Monitoring	Forests are monitored 24/7 by the web camera
5.	Detection	The system shall take training sets of fire and check for fire or no fire.
6.	Alert Message	The system will send notification to the authorities when fire is detected

Non-Functional Requirements:

S.No	Non-Functional Requirement	Description
1.	Usability	The forest fire can be detected as fast as possible.
2.	Security	A huge area of forest can be saved from forest fires
3.	Reliability	The Web camera that uses CNN detects the clear image of fire.
4.	Performance	Accurately measures the radius of fire that is being spread while the forest fire occurs with the help of web cameras. This enables authorities arrange the man power accordingly.
5.	Availability	At the correct time, the authorities will receive the alert from the web cameras.
6.	Scalability	If the authorities take quick action once they receive the alert message, they will be able to prevent huge loss of forest area.

PROJECT DESIGN

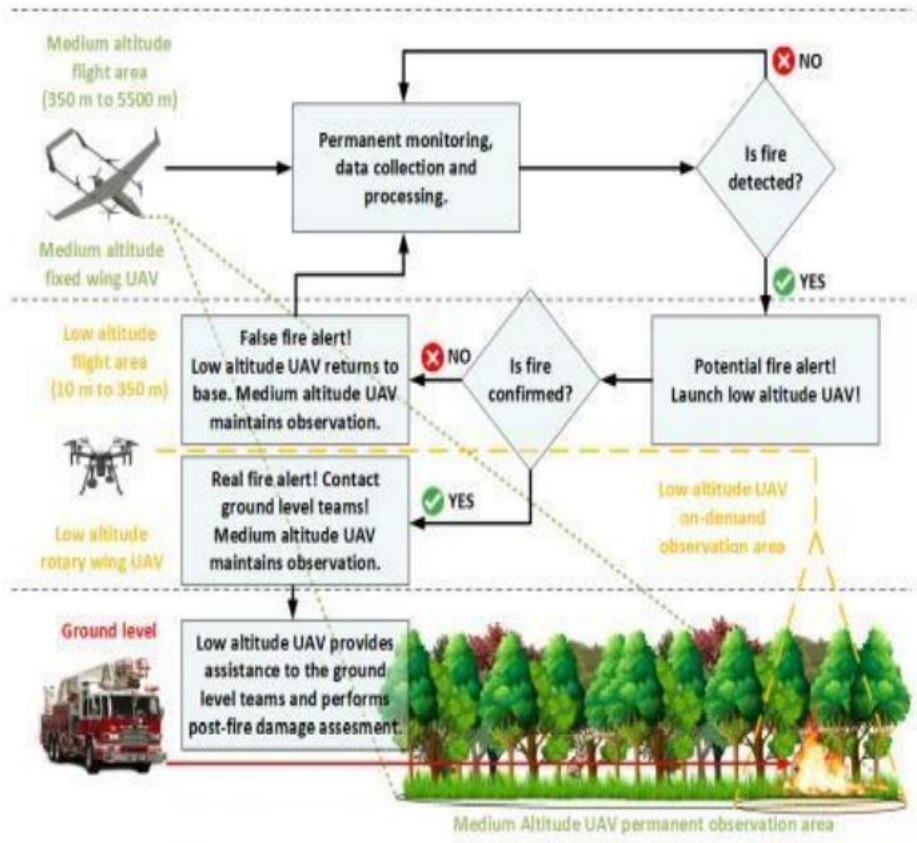


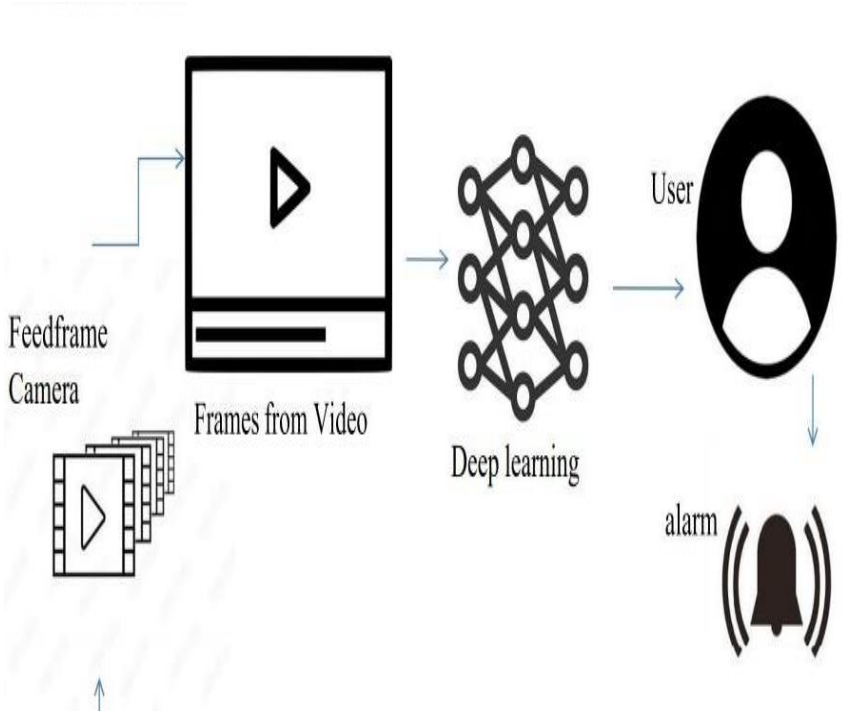
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system.

A neat and clear DFD can depict the right amount of the system requirement graphically.

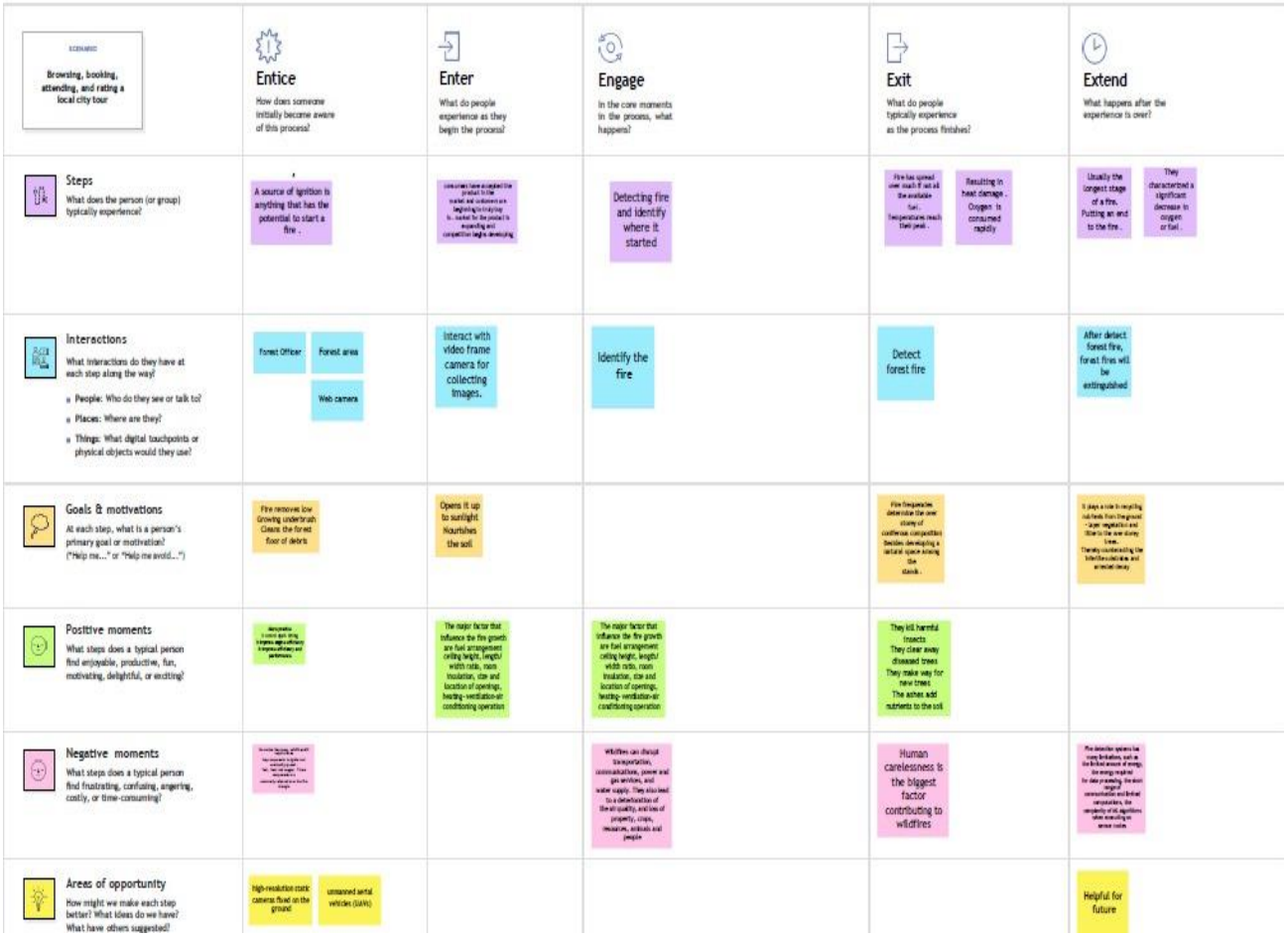
It shows how data enters and leaves the system, what changes the information, and where data is stored.

SOLUTION & TECHNICAL ARCHITECTURE





CUSTOMER JOURNEY



Use this framework to better understand customer needs, motivation, and obstacles by illustrating a key scenario or process from start to finish. When possible, use this map to document and summarise interviews and observations with real people rather than relying on your assumptions.

PROJECT PLANNING AND SCHEDULING

Product Backlog, Sprint schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	20	High	All members
		USN-2	As a user, I will receive confirmation email once I have registered for the application usage.	20	High	All members
Sprint-2	Input	USN-3	Whenever the fire is detected, the information is given to the database.	20	High	All members
Sprint-2		USN-4	When it is the wildfire then the alarming system is activated.	20	High	All members
Sprint-3	Output	USN-5	And the alarm also sent to the corresponding departments and made them know that the wildfire is erupted.	20	High	All members
Sprint-4	Action	USN-6	Required actions will be taken in order to control erupted wildfire by reaching as early as possible to the destination with the help of detecting systems.	20	High	All members

Project Tracker, Velocity and Burndown Chart

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

CODING AND SOLUTIONING

Python code:

1. Model building
2. Open CV for video processing
3. Training Image classification (in IBM cloud Watson Studio)

2.Initializing the model

In [9]:
```
model=Sequential()
```

3.Adding Cnn Layers

In [10]:
```
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
```

4.Adding Dense layers

In [11]:
```
model.add(Dense(150,activation='relu'))
model.add(Dense(1,activation='sigmoid'))
model.add(Dense(5,activation="softmax"))
```

5.Configuring the learning process

In [12]:
```
model.compile(loss='binary_crossentropy',optimizer="adam",metrics=["accuracy"])
```

7.Save the model

In [17]:
```
model.save("forest1.h5")
```

8.Predicate

In [18]:
```
#import load_model from keras.model
from keras.models import load_model
#import image class from keras
from tensorflow.keras.preprocessing import image
#import numpy
import numpy as np
#import cv2
import cv2
```

In [19]:
```
model = load_model("forest1.h5")
```

In [20]:
```
img=image.load_img('/content/drive/MyDrive/ibm/fire/dataset/test_set/with fire/with fire (3).png')
z=image.img_to_array(img)
res = cv2.resize(z, dsize=(128, 128), interpolation=cv2.INTER_CUBIC)
#expand the image shape
z=np.expand_dims(res,axis=0)
```

In [21]:
```
pred=model.predict(z)
```

```
1/1 [==============================] - 0s 76ms/step
```

In [22]:
```
pred
```

Out[22]:
```
array([[1.]], dtype=float32)
```

In [36]:
```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='9N-Z9zqtcU1AMs9XRJUQyjtMOxisjuGS3vxfLJi4rIoR',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'trainimageclassification-donotdelete-pr-bqvosr2xvgpyrh'
object_key = 'forest fire.zip'

streaming_body_3 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
```

In [37]:
```
from io import BytesIO
import zipfile
unzip = zipfile.ZipFile(BytesIO(streaming_body_3.read()),'r')
```

In [29]:
```
model_details = client.repository.store_model(model= "image-classification-model_new.tgz",meta_props={
client.repository.ModelMetaNames.NAME:"CNN",
client.repository.ModelMetaNames.TYPE:"tensorflow_2.7",
client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid}
                                                    )
model_id = client.repository.get_model_id(model_details)
```

In [30]:
```
model_id
```

Out[30]:
```
'05d3cd53-1f1f-487c-917a-8d1ac8d18c2b'
```

```
account_sid = 'ACb0eefb4e6364ae26e9591b2a756bbc8f'
auth_token = '531ce9cfe612fd23c7691d644a0ffe26'
client = Client(account_sid, auth_token)

message = client.messages \
    .create(
         body='Forest fire is detected , stay alert',
         from_='+13465507864',
         to='+917397487445'
     )

print(message.sid)
```

```
SM3fcb166333c7514243fbdf26fb158a19
```

In [40]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                   "url": "https://us-south.ml.cloud.ibm.com",
                   "apikey": "CJDjIodTlDY0KrQmB15hWQQLi9Dbzss8VDAfeFrqLU8c"
                  }
client = APIClient(wml_credentials)
```

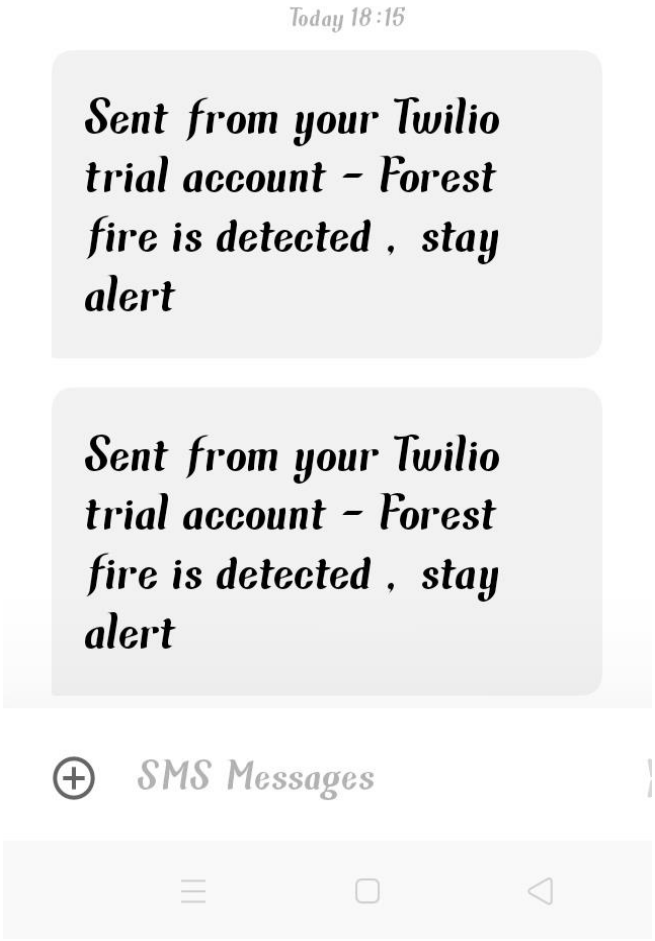
In [23]:
```
client = APIClient(wml_credentials)
```

In [24]:
```
def guid_from_space_name(client, space_name):
    space = client.spaces.get_details()
    #print(space)
    return(next(item for item in space['resources']if item['entity']["name"] == space_name)['metadata']['id'])
```

In [25]:
```
space_uid = guid_from_space_name(client, 'imageclassification')
print("Space UID = " + space_uid)
```

RESULT

PERFORMANCES MATRICES:



.

CONCLUSION

In this paper we have briefly presented two new methods for early forest fire detection, including part of their characteristics and main components. We have also analysed some of the benefits, which these methods can provide to the involved Bachelor, Master and PhD students. Both solutions are still under development, but they show great potential and work on their development and improvement will continue in the following years.

FUTURE SCOPE

Evolution emerges in the processing, computation, and algorithms. This strives many researchers to pay attention in many domains where they work in the processing of surveillance video streams so that abnormal or unusual actions could be detected. The usage of UAVs is recommended in the detection of forest fire due to the high mobility and ensures the coverage areas at various altitudes and locations at a low cost. Hence, an efficient and scalable UAV is used for detection. This work aims in developing the 3D model for the captured scene. YOLOv4 tiny network is deployed to detect the fire. The accuracy of the detection rate achieved through this model is 91%. The proposed model outperforms the other existing techniques in terms of detecting in the early stage. However, this model is sensitive to the forest with dense fogs and clouds.

APPENDIX

1. https://github.com/IBM-EPBL/IBM-Project-22684-1659856274
2. https://cloud.ibm.com/
3. https://console.twilio.com/?frameUrl=%2Fconsole%3Fx-target-region%3Dus1
4. https://careereducation.smartinternz.com/student-enroll-login
5. https://www.kaggle.com/arbethi/forest-fire?select=Dataset
6. https://colab.research.google.com/drive/1mAnJnf6fpjlG8KVez5tMxxKDe_OGB9d8#scrollTo=uLL6iBRnSCl7