SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Computing Technologies
CTECH
Accelerate Intellectual Innovation

SRM Institute of Science and Technology
School of Computing

FIRE DETECTION USING CONVOLUTION NEURAL NETWORK(CNN)

ABSTRACT

This project presents a deep learning-based solution for automated fire detection using Convolutional Neural Networks (CNNs). Utilizing publicly available fire and smoke image datasets, the system processes visual input to classify whether fire is present in an image. Image preprocessing techniques, data augmentation, and a custom CNN architecture contribute to model robustness. A Streamlit web app provides real-time image upload and classification. The system aims to support early fire detection in surveillance and safety-critical environments.

OBJECTIVES

- Build an image classification model using CNN for detecting fire.
- Develop a real-time interactive web app using Streamlit.
- Ensure high accuracy and fast inference for practical deployment.
- Provide visual feedback such as prediction labels and confidence scores.

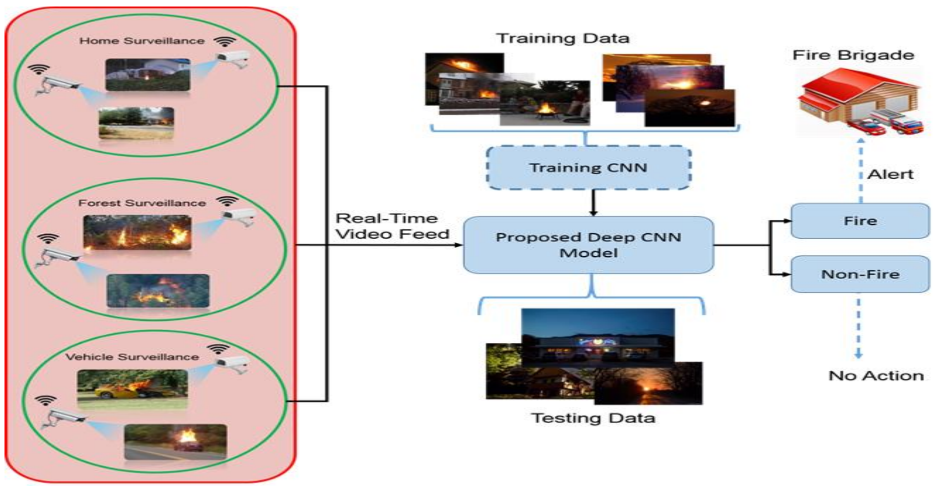
ANALYSIS

- Data Collection and Exploration- uses databases from kaggle.
- Image Preprocessing- Data Augmentation
- Model training and saving
- Model Evaluation
- Web Application

PERFORMANCE METRICS

Accuracy: ~ 98%
Response Time:~1.5 seconds per image
Memory Usage: ~1.2GB RAM
Uptime: 99%
Robustness: Works well on unseen fire condition

ARCHITECUTRE DIAGRAM :--



STEPS

i.Dataset Collection
ii.Preprocessing and Augmentation
iii.Model Design(CNN)
iv.Training and evaluation
v.Saving and Loading model
vi.Deploying via Streamlit
vii.Creating public link using PyNgrok

METHODOLOGY

NN Model Example:
python
Copy
Edit

```
model = Sequential([
  Conv2D(32, (3,3), activation='relu', input_shape=(224, 224, 3)),
  MaxPooling2D(2,2),
  Conv2D(64, (3,3), activation='relu'),
  MaxPooling2D(2,2),
  Flatten(),
  Dense(128, activation='relu'),
  Dense(2, activation='softmax')  # fire, no fire
])
```

Streamlit for Web App
Keras/TensorFlow for model training
PyNgrok for public deployment

RESULTS

Test accuracy: ~96%

Average prediction time: < 2 seconds

Real-time image upload and detection

Successfully detects fire in varied lighting and backgrounds

Model is lightweight and suitable for edge deployment

FUTURE ENHANCEMENTS

Integration with CCTV feed:

Real time surveillance detection.

Mobile App Deployment: Use TensorFlow Lite

TEAM MEMBERS

RA2311003011644-TANMAY KUMAR SINGH
RA2311003011639-DIVYA BANSAL
RA2311003011643-HIMANSHU JHA