Pothole Pathfinder

K.Renu Sreeja , K.Nimisha , A.Shivani , K.Sahithi

Under the esteemed guidance of

Dr. P.Kayal

Associate Professor

VISHNU
UNIVERSAL LEARNING

Bachelor of Technology
Department of Information Technology
BVRIT HYDERABAD College of Engineering for Women

December 3, 2023

Overview

Introduction

- Deep learning enables real-time pothole detection.
- Interactive dashboard supports efficient road maintenance and hazard identification.
- Pothole detection systems enhance road safety and transportation efficiency.

Problem Statement

- Create a cost-effective, real-time pothole detection system using deep learning to counter safety risks and traffic disruptions caused by conventional, slow methods.
- The project employs deep learning algorithms for real-time pothole identification from video and images, including depth and width analysis.
- Robust evaluation metrics in model training and testing ensure system reliability and accuracy, with data visualized on an interactive dashboard.

Literature Survey

S. No	Title of the paper	Author(s) & Journal Details	Description
1	AI assisted pothole detection and depth estimation	Eshta Ranyal, Ayan Sadhu, Kamal Jain - 2023	This paper introduces an advanced RetinaNet CNN for efficient 3D vision-based pothole detection. High F1 scores and 5% mean error in benchmark dataset validation promise cost-effective road safety and pavement preservation.

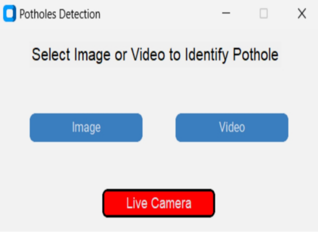
S. No	Title of the paper	Author(s) & Journal Details	Description
2	Real-time machine learning-based approach for pothole detection	Oche Alexander Egaji , Gareth Evans , Mark Graham Griffiths , Gregory Islas - 2021	This paper compares pothole detection models using diverse data sources and 2-second non-overlapping window preprocessing. Cross-validation yielded an accuracy of 0.8889 for Random Forest Tree and KNN. After hyperparameter tuning, Random Forest Tree achieved an accuracy of 0.9444, showing strong precision, recall, and F-score.

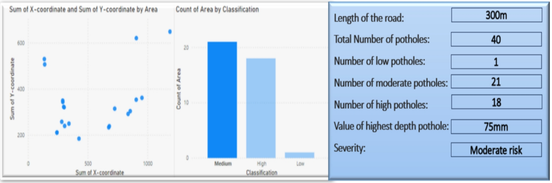
S. No	Title of the paper	Author(s) & Journal Details	Description
3	An Automated Machine-Learning Approach for Road Pothole Detection Using Smartphone Sensor Data	Chao Wu ,Zhen Wang,Simon Hu ,Julien Lepine - 2020	This paper utilizes smartphone sensors to gather road condition data, employing machine learning to classify defects. It's cost-effective, efficient, and offers real-time insights, yet faces constraints in dataset size and data acquisition variability.

S. No	Title of the paper	Author(s) & Journal Details	Description
4	Pothole Detection and Volume Estimation using Stereoscopic Cameras	Margaret Velse Thekkethala, Reshma S, Sebin Jacob Varughese, Vaishnavi Mohan, Geevarghese Titus - 2016	The method uses dual perspectives (left and right) to capture potholes, employs diverse image operations for isolation, applies 3D mapping for volume calculation, achieving an impressive 83% detection accuracy. (or) This method uses dual pothole perspectives, applies diverse image processing, and employs 3D mapping for 83% accurate pothole detection.

Proposed System

- **Pothole Detection:** Our system employs YOLOv4 for precise pothole detection.
- **Area Calculation:** It calculates pothole areas using width and height information from detection.
- **Depth Estimation:** Monocular depth estimation is used to assess the depth of each pothole accurately.
- **Interactive Dashboard:** Integrating pothole insights into interactive dashboard for users.





Expected Dashboard

Tools and Technology

Hardware Specifications

- Processor - Intel Core i5
- Memory(RAM) - 8 GB
- Storage - 1TB

Software Specifications

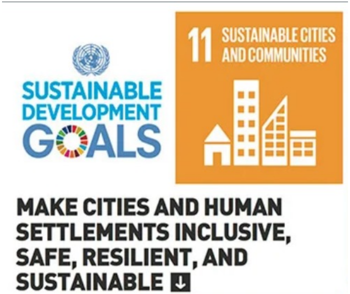
- Python 3.9 and later versions
- OS - Windows 10 and later
- Jupyter Notebook

Feasibility Study

- For image and video processing we had employed OpenCV.
- For pothole detection we employed YoloV4 deep learning algorithm.
- We are planning to use machine learning or RetinaNet CNN for pothole depth classification.
- For real time road analytics we will be accessing Tableau dashboard.

Societal Impact

- **Goal 11:** Sustainable Cities and Communities promotes sustainable urban development, encompassing efficient transportation and road maintenance, which can help prevent pothole issues. This goal's focus on inclusive, safe, and sustainable cities indirectly reduces the need for extensive pothole detection and repairs.



Partial Implementation

- GUI Interface

```
    if len(filename) > 0:
        showinfo(title='Selected Video File', message=filename)
        # Call the detectPotholeonVideo function and get the pothole data
        pothole_data_list = videoDetector.detectPotholeonVideo(filename)
        # Check if there is pothole data
        if pothole_data_list:
            showinfo(title='Pothole Detection', message='Pothole data has been saved to pothole_data.csv')

# Image open button
image_open_button = customtkinter.CTkButton(root, text='Image', command=select_image_file, hover_color="green")

# Video open button
video_open_button = customtkinter.CTkButton(root, text='Video', command=select_video_file, hover_color="green")

# Live Camera button
liveCamera_button = customtkinter.CTkButton(root,
                                            text='Live Camera',
                                            command=lambda: videoDetector.detectPotholeonVideo(0),
                                            hover_color="green",
                                            border_color="black",
                                            border_width=2.5,
                                            fg_color="red", font=("poppins", 14))

HeadingText.place(x=50, y=15)
image_open_button.place(x=40, y=80)
video_open_button.place(x=220, y=80)
liveCamera_button.pack(side='bottom', pady=20)
```

- VideoDetector.py

```python
def detectPotholeonVideo(filename):
    # Create an empty set to store unique areas
    unique_areas = set()

    # Create an empty list to store pothole data
    pothole_data_list = []

    # reading label name from obj.names file
    class_name = []
    with open(os.path.join("project_files", 'obj.names'), 'r') as f:
        class_name = [cname.strip() for cname in f.readlines()]

    # importing model weights and config file
    # defining the model parameters
    net1 = cv.dnn.readNet('project_files/yolov4_tiny.weights', 'project_files/yolov4_tiny.cfg')
    net1.setPreferableBackend(cv.dnn.DNN_BACKEND_CUDA)
    net1.setPreferableTarget(cv.dnn.DNN_TARGET_CUDA_FP16)
    model1 = cv.dnn_DetectionModel(net1)
    model1.setInputParams(size=(640, 480), scale=1 / 255, swapRB=True)

    # defining the video source (0 for camera or file name for video)
    cap = cv.VideoCapture(filename)
    width = cap.get(3)
    height = cap.get(4)
    result = cv.VideoWriter('PotholeVideoResult.mp4',
                            cv.VideoWriter_fourcc(*'MP4V'),
                            10, (int(width), int(height)))
```

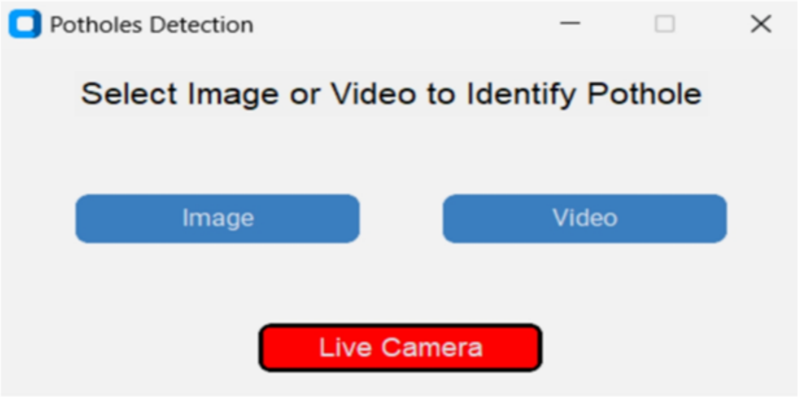
- ImageDetector.py

```python
def detectPotholeonImage(image_filename):
    # Create an empty list to store pothole data
    pothole_data_list = []

    # Reading label names from obj.names file
    class_name = []
    with open(os.path.join("project_files", 'obj.names'), 'r') as f:
        class_name = [cname.strip() for cname in f.readlines()]

    # Importing model weights and config file
    net1 = cv.dnn.readNet('project_files/yolov4_tiny.weights', 'project_files/yolov4_tiny.cfg')
    net1.setPreferableBackend(cv.dnn.DNN_BACKEND_CUDA)
    net1.setPreferableTarget(cv.dnn.DNN_TARGET_CUDA_FP16)
    model1 = cv.dnn_DetectionModel(net1)
    model1.setInputParams(size=(640, 480), scale=1 / 255, swapRB=True)

    # Load the image
    frame = cv.imread(image_filename)

    # Process the image
    classes, scores, boxes = model1.detect(frame, confThreshold=0.1, nmsThreshold=0.4)

    for (classid, score, box) in zip(classes, scores, boxes):
        label = "Pothole"
        x, y, w, h = box
        area = w * h  # Calculate the area of the bounding box
```

Results & Discussion

Our proposed system introduces a user-friendly graphical interface that provides three distinct options for processing: images, videos, or live camera input as shown in Fig 5.4. Users can seamlessly choose their preferred mode of anal- ysis. Shown in Fig 5.5, 5.6, our system leverages YOLOv4 for robust pothole detection, the system calculates dimensions and estimates depth using the Mi- DaS model, with due consideration to frames per second (fps). Real-time results are displayed, offering immediate insights, while all pertinent pothole details are meticulously recorded in a CSV file. This systematic data storage facilitates fur- ther processing and in-depth analysis.

- Pothole Severity Prediction



- Pothole Detection with confidence score 0.81

- Pothole Detection with confidence score 0.99

Status of Project Report and Publication

- Submitted and verified report from our guide
- Submitted a handwritten copy to our guide

References

- Eshta Ranyal, Ayan Sadhu, Kamal Jain. 2023. "AI assisted pothole detection and depth estimation". In the International Conference on Machine Intelligence for GeoAnalytics and Remote Sensing (MIGARS). DOI : 10.3390/app112311229.
- Oche Alexander Egaji , Gareth Evans , Mark Graham Griffiths , Gregory Islas. 2021. "Real-time machine learning-based approach for pothole detection". Published by Elsevier Ltd. DOI:10.1016/j.eswa.2021.115562.
- Chao Wu ,Zhen Wang,Simon Hu ,Julien Lepine. 2020. "An Automated Machine-Learning Approach for Road Pothole Detection Using Smartphone Sensor". DOI : 10.3390/s20195564
- Margaret Velse Thekkethala, Reshma S, Sebin Jacob Varughese, Vaishnavi Mohan, Geevarghese Titus. 2016. "Pothole Detection and Volume Estimation using Stereoscopic Cameras". In the International Journal of Industrial Electronics and Electrical Engineering. ISSN no : 2347-6982. Volume-4, Issue-5, May.-2016.

Thank you