A COMPREHENSIVE REVIEW OF POTHOLE DETECTION TECHNIQUES UTILIZING IMAGE OR VIDEO FEED FOR ROAD MAINTENANCE AND SAFETY

BM SOMASHEKAR , SHREYAS H S , G SAI ROOPESH, HEMANTH KUMAR S, ANJAN S S

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING,INDIA

[1dt21ai008@dsatm.edu.in](mailto:1dt21ai008@dsatm.edu.in), [1dt21ai052@dsatm.edu.in](mailto:1dt21ai052@dsatm.edu.in) ,[1dt21ai019@dsatm.edu.in](mailto:1dt21ai019@dsatm.edu.in) ,[1dt21ai025@dsatm.edu.in](mailto:1dt21ai025@dsatm.edu.in) [1dt22ai400@dsatm.edu.in](mailto:1dt22ai401@dsatm.edu.in)

**ABSTRACT**

Potholes on road surfaces are a major hazard to road safety, infrastructure upkeep, and traffic efficiency. Pothole detection and prompt repair are critical for reducing potential accidents and repair costs. Advances in computer vision and deep learning techniques in recent years have paved the path for the development of automated systems capable of spotting potholes in real-time utilizing live video feeds from roadways. This study of the literature investigates cutting-edge methodologies and research contributions in the field of pothole identification using Convolutional Neural Networks (CNNs) applied to live video data.

**Keywords**

Pothole Detection,Real-Time Detection,Convolutional Neural Networks,Computer Vision.

**1. INTRODUCTION**

**1.1. Pothole Detection**

The detection and quick repair of potholes on road surfaces is a top priority for transportation agencies worldwide. These road defects not only endanger drivers but also cause costly infrastructural damage. This literature study goes into the delicate domain of pothole detection as part of the larger landscape of intelligent transportation systems and computer vision applications.

This review's primary focus is on the development and application of computer vision techniques, specifically Convolutional Neural Networks (CNNs), for the automatic detection of potholes in real-time scenarios.

This article goes into the complex world of pothole detection by investigating several CNN-based architectures, data pretreatment methodologies, and the importance of balanced datasets in training robust models. Furthermore, it delves into the various issues of real-world testing and assessment criteria that are crucial for evaluating the efficacy of pothole detecting systems.As road networks develop and evolve, the deployment of scalable, adaptive, and cost-effective pothole

detection solutions become critical. This review is a comprehensive resource for researchers, politicians, and transportation authorities interested in leveraging the promise of computer vision and deep learning approaches to produce

safer and more efficient transportation networks. This paper contributes to the broader discussion of intelligent infrastructure management and road safety enhancement by focusing on the critical problem of pothole identification.

**1.2. Real Time Detection-**

In the realms of road safety and infrastructure management, real-time pothole detection is a game changer." The code under consideration, as mentioned in the literature review report, focuses a strong emphasis on 'Real-Time' detection. This crucial feature highlights the code's capacity to quickly and effectively identify potholes as they appear in live video feeds of roadways.

Real-time detection is critical in an era when urbanization and increased vehicular traffic have exacerbated the issues connected with road maintenance. The ability to detect and report potholes in real time not only helps to prevent accidents and vehicle damage, but it also allows for proactive infrastructure maintenance.

The real-time capability of the code was developed in response to the insufficiency of traditional manual inspections, which are time-consuming and ineffective in addressing developing road risks. The programme aims to provide a viable and automated solution to this critical problem by utilizing cutting-edge Convolutional Neural Networks (CNNs) and image processing techniques.

This review study looks into the complexities of real-time pothole identification, investigating the technical foundations, data preparation approaches, and the delicate balance of datasets for model training. It also examines the intricacies of evaluating detection algorithms in real-world circumstances, emphasizing the code's contributions to addressing these issues.

.

**2.Methodologies**

**2.1. Data Collection**

The dataset for pothole identification in this study was obtained from Kaggle, a popular site for sharing and hosting datasets. The dataset, suitably named the "Pothole Detection Dataset," was painstakingly assembled with the goal of developing a pothole detecting model. The following are the major elements of data collection:

**Data Source**:

The dataset was generated as part of a college project to construct a pothole detecting model by the dataset owner. Images were gathered from numerous websites and meticulously arranged into two categories: "normal" and "potholes."

**Dataset Organization**:

The dataset is divided into two main folders: "normal" and "potholes." The "normal" collection contains photographs of smooth roads taken from various angles and views. The "potholes" folder, on the other hand, comprises photos of road surfaces with potholes, illustrating the difficult scenarios faced in real-world road conditions.

**Purposeful Curation**:

The dataset was rigorously curated to include a wide range of road conditions, ensuring that it is indicative of the issues that pothole detection models confront in real-world scenarios.

While the dataset was obtained via the internet, ethical considerations were taken into account to ensure compliance with data usage regulations and standards. The dataset contains no sensitive or personal information.

**Accessibility**:

The dataset is now publically available on Kaggle, “https://www.kaggle.com/datasets/atulyakumar98/pothole-detection-dataset”, making it easier for researchers and amateurs interested in developing pothole detection models to use.

****

**Fig1. Training Set(Plain.jpg)**

****

**Fig2. Training Set(Pothole.jpg)**

**2.2. Data Preprocessing**

Some the data preprocessing techniques used for pothole detection in the model are:

**Image Scaling:**

Using OpenCV's cv2.resize() method, all images are enlarged to a consistent dimension (e.g., 300x300 pixels). This step ensures that image proportions are consistent.

**Normalization:**

The image pixel values are normalized by dividing them by 255.0. This step scales pixel values to a range between 0 and 1, which is a frequent deep learning practice to facilitate convergence.

**Data Augmentation** :

To boost the diversity of the training dataset, data augmentation techniques such as random rotation, scaling, and horizontal flips are frequently used. Although it is not explicitly demonstrated in this model, data augmentation is a typical practise in deep learning for picture classification problems

.

**Label Encoding**:

Pothole and non-pothole photos have labels encoded as 1 and 0, respectively. This stage turns category labels into numerical data that may be used to train a machine learning model.

**Shuffling:**

shuffle() from scikit-learn is used to shuffle the training and test datasets. Shuffling helps to randomize the order of data points, which is useful for properly training neural networks.

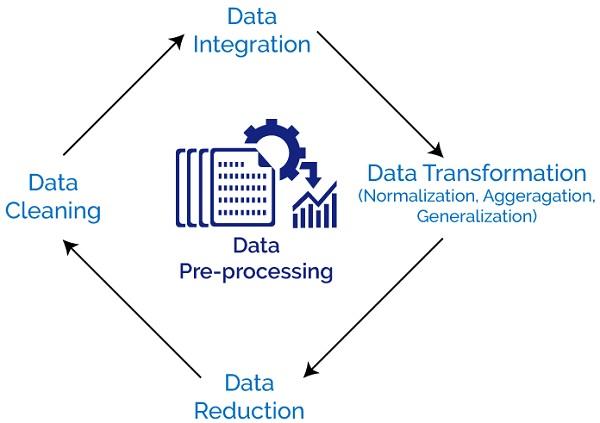
**Label Binarization (One-Hot Encoding):**

The categorical labels are encoded using one-hot encoding for the classification job.

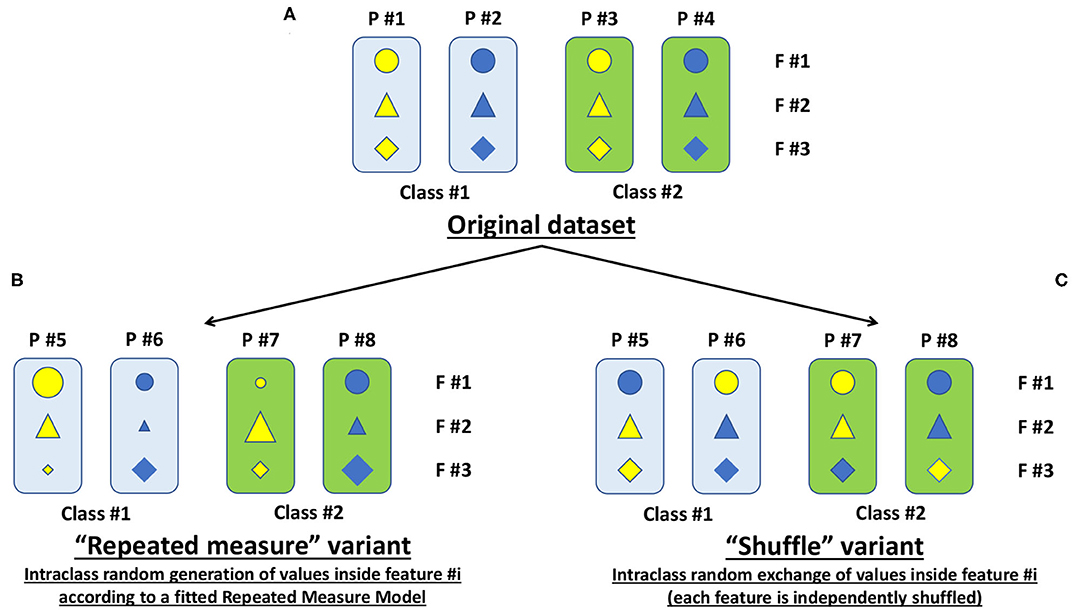
Keras' to\_categorical() function is used to convert class labels to binary matrix representation.

These preprocessing approaches are critical for preparing the dataset and ensuring that it is fit for effective training of the convolutional neural network (CNN) model.

We may also include  **hyperparameter tuning** in this model ,particularly for optimizing the performance of pothole detection models.as it is a crucial step in the development of effective deep learning models and is often performed in practice to achieve the best results.

****

**Fig3. A Typical Flow of Data Preprocessing**

****

**Fig4. Implementation of Shuffling**

**2.3. Convolutional Neural Network (CNN)**

Convolutional neural network (CNN) is a [regularized](https://en.wikipedia.org/wiki/Regularization_(mathematics)) type of [feed-forward neural network](https://en.wikipedia.org/wiki/Feed-forward_neural_network) that learns [feature engineering](https://en.wikipedia.org/wiki/Feature_engineering) by itself via [filters](https://en.wikipedia.org/wiki/Filter_(signal_processing)) (or kernel) optimization. Vanishing gradients and exploding gradients, seen during [backpropagation](https://en.wikipedia.org/wiki/Backpropagation) in earlier neural networks, are prevented by using regularized weights over fewer connections. For example, for each neuron in the fully-connected layer 10,000 weights would be required for processing an image size 100x100 pixels. However, applying cascaded convolution (or cross-correlation) kernels only 25 neurons are required to process 5x5-sized tiles.Higher-layer features are extracted from wider context windows, compared to lower-layer features

**Role of CNNs in Pothole Detection**

**Feature Extraction**

Convolutional Neural Networks (CNNs) have gained immense popularity in computer vision tasks due to their unparalleled ability to automatically extract meaningful features from images. In the context of pothole detection, this feature extraction capability is crucial.

**Pattern Recognition**

Potholes, like many objects, are characterized by distinct patterns and shapes. CNNs excel at pattern recognition, making them adept at distinguishing potholes from their surroundings.

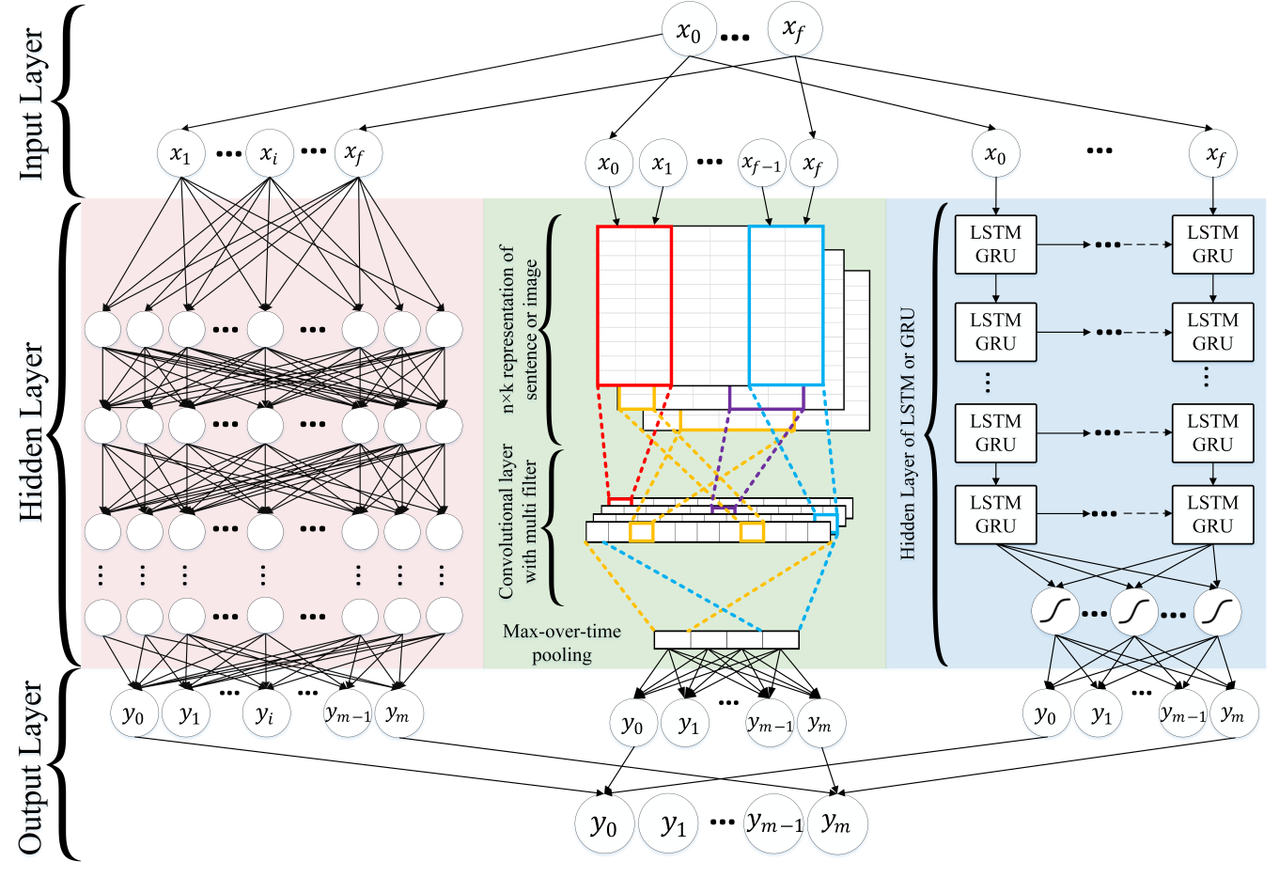
CNNs can identify subtle differences in texture, color, and shape that distinguish potholes from regular road surfaces. They can automatically learn to recognize these discriminative patterns during training.

**Adaptability and Generalization**

CNNs are capable of adapting to changing road conditions, lighting, and camera angles. They generalize effectively, which means they can detect flaws in real-world settings outside of the training dataset.

**CNN Architecture**

A convolutional neural network consists of an input layer, [hidden layers](https://en.wikipedia.org/wiki/Artificial_neural_network#Organization) and an output layer. In a convolutional neural network, the hidden layers include one or more layers that perform convolutions. Typically this includes a layer that performs a [dot product](https://en.wikipedia.org/wiki/Dot_product) of the convolution kernel with the layer's input matrix. This product is usually the [Frobenius inner product](https://en.wikipedia.org/wiki/Frobenius_inner_product), and its activation function is commonly [ReLU](https://en.wikipedia.org/wiki/Rectifier_(neural_networks)). As the convolution kernel slides along the input matrix for the layer, the convolution operation generates a feature map, which in turn contributes to the input of the next layer. This is followed by other layers such as pooling layers, fully connected layers, and normalization layers.



**Fig5. Structure of a Typical Neural Network**

**Real-Time Processing**

CNNs' adaptability for real-time processing is one of their primary advantages. They can analyze live video feeds quickly, making them crucial for detecting potholes on the road.CNNs process video frames quickly, allowing for instantaneous notifications or actions in reaction to pothole detection.

Firstly, by specifying the input shape in the first convolutional layer, the input layer is implicitly defined. It accepts grayscale images with an input size of (size, size, 1), where size is 300 in this instance.

**Convolutional layer 1, or layer one-**

Conv2D(16, 8 8), strides=4 4 and padding="valid",16 filters of size (8, 8) are applied to the input by this layer.

The convolution operation utilises a stride of (4, 4), which indicates that the filter travels 4 pixels at a time.

No zero-padding is added to the input, resulting in a smaller output volume and "valid" padding.

ReLU (Rectified Linear Unit), the activation function in use, adds nonlinearity to the mode.

**Convolutional layer two**-

Padding="same" Conv2D(32, (5, 5)).The output of the preceding layer is subjected to 32 filters of size (5, 5) in this layer. In order to guarantee that the output has the same spatial dimensions as the input, it utilises "same" padding. ReLU is once again utilised as the activation function.

**Average Pooling Layer-**

GlobalAveragePooling2D(). The average value for each feature map across all spatial dimensions is calculated for this layer.

It flattens the feature maps by reducing the spatial dimensions to 1x1.

**Dense (Fully Connected) Layer:**

**Dense(1)**

512 neurons make up this layer. By activating the ReLU, it adds a new nonlinearity. Applying dropout at a rate of 0.1 prevents overfitting by changing a portion of the input units randomly to 0 during each update.

**Result Layer:**

**Dense(2)**

Two neurons make up the output layer, one for each of the two classes—"pothole" and "non-pothole." Softmax is the activation function, and it transforms the model's unprocessed output into class probabilities.

**2.4. Training Strategy**

Here we should address the following crucial points when describing the training technique employed in the model.

**Preparation of Data**:

Here we look into the process used to create the training dataset. In this instance, it entails gathering pictures of both potholes and smooth surfaces.

Next we look into data augmentation methods that were utilized, such as scaling photographs to a standard size (such as 300x300 pixels) and normalizing pixel values to a [0, 1] range.

**Architectural models**

Here we describe the CNN architecture that is utilized to identify potholes. In your situation, the kerasModel4 function defines the architecture. Convey information on the number of layers, the different types of layers (convolutional, pooling, and dense), and the activation mechanisms employed in each layer.

**Training Methodology:**

Categorical cross-entropy loss is employed in this model and is appropriate for a multi-class classification problem.

The number of samples processed in each training step is determined by the batch size that was utilized during training.

Epochs: Number of the training epochs, which represent the number of times the complete training dataset is processed.

There was also the portion of the training data that was set aside for validation, if any, in order to keep track of the model's progress.

**Training Cycle**

We can employ the process of shuffling the training data to ensure that each batch has a varied range of samples and the process used to normalize the data (scale pixel values to [0, 1]) before incorporating it into the model.

**Evaluation Criteria**

In this particular model we used the parameter as accuracy, which assesses the model's capacity to accurately distinguish potholes and non-potholes, as the major statistic utilized for evaluation.

**Preventing overfitting**

To avoid overfitting, dropout layers are introduced to the model with a predetermined dropout rate (for example, 0.1).

The dropout forces the model to acquire more robust features by randomly deactivating a portion of neurons during training.

**Training Methodology:**

The kerasModel4 function in the code contains a definition of the training procedure.

The Adam optimizer and categorical cross-entropy loss, which are appropriate for multi-class classification problems like pothole detection, are used in the model's construction.

**Training Cycle:**

The model is trained on the training dataset using the model.fit method, which starts the training loop.

Prior to training, the training data is jumbled and standardized.

**Batch size and epochs:**

The model.fit method specifies the batch size and the number of training epochs.

50 epochs are mentioned in this scenario for faster model training but you can increase the number of epochs for better accuracy of the model..

A part of the training data, in this model its 10%, is set aside for validation using the validation\_split option.

**Monitoring Training Development:**

Each epoch includes a monitoring of the training progress.

At the conclusion of each epoch, the history object holds metrics like training and validation accuracy and loss.

**Getting to Training Metrics:**

The history object allows access to the metrics, such as accuracy and loss for both training and validation.

For instance, you may use history.history['accuracy'] and history.history['loss'] to retrieve the training accuracy and loss.

Similar to this, you may use history.history['val\_accuracy'] and history.history['val\_loss'] for validation accuracy and loss.

**Progress tracking for training:**

The training metrics are not explicitly logged here, but logging statements may be simply added to report the metrics at the conclusion of each epoch.

By altering the training loop, for instance, you may report the training accuracy and loss after each epoch.

**Performance Patterns or Modifications:**

The trends in training and validation measures over epochs may be analyzed to spot patterns or changes in performance throughout training.Indicators that the model is learning from the training data include an improvement in training accuracy and a decrease in training loss over epochs.a consistent pattern in validation accuracy and loss, demonstrating the model's ability to generalize to new data.

Validation metrics that fluctuate or suddenly plummet might be an indication of overfitting.

Metric plateaus or a trend toward gradual convergence suggest that the performance of the model has stabilized.

**2.5 Evaluation Metrics**

In this model, the evaluation metrics are used to assess how well the CNN model identifies potholes. They are routinely used to measure the performance of a machine learning model.

The model's training accuracy is calculated and shown using this code. One of several evaluation indicators is training accuracy.

For a thorough analysis, you can think about incorporating other metrics in your research, such as precision, recall, F1-score, and confusion matrix. The meaning of these measurements is as follows:

The percentage of accurately predicted samples in the training dataset is known as training accuracy.

It gauges how effectively the model has internalized the training set.

The calculation is (samples properly categorized) / (samples used for training) \* 100.

High training accuracy is a sign of effective learning, but it doesn't always mean that the model generalizes well to new data.

You may take into account the following indicators to aid your paper's review to be more thorough:

**Testing Accuracy**:

Similar to training accuracy but calculated on the testing dataset. Measures the model's ability to make accurate predictions on unseen data.

The formula is (correctly classified samples) / (total testing samples) \* 100.

**Precision:**

Precision measures the ability of the model to correctly identify positive samples (e.g., potholes) among all samples predicted as positive.

High precision indicates that when the model predicts a positive class, it's usually correct.

The formula is (true positives) / (true positives + false positives).

**Sensitivity (Recall)**:

Recall gauges a model's capacity to accurately distinguish each positive sample from each genuine positive sample.

High recall means that a significant fraction of the real positive samples may be captured by the model.

(True Positives) / (True Positives + False Negatives) is the equation.

**F1-Score**:

The harmonic mean of recall and accuracy is known as the F1-score.

It offers a fair evaluation of a model's effectiveness by taking into account both false positives and erroneous negatives.

2 \* (precision \* recall) / (precision + recall) is the formula.

Uncertainty Matrix:

A table that lists the model's predictions and actual class labels is known as a confusion matrix.

True positives, true negatives, false positives, and false negatives are all covered in detail.

It's helpful to know which classes are frequently misunderstood and where the model makes mistakes.

The performance of the pothole detection model will be evaluated in your work more thoroughly and in-depthly if you provide these evaluation measures.

**2.6 Real Time Prediction**

A crucial application that has practical uses for road maintenance and road safety is pothole detection using live video feed. A live video stream from a camera mounted on a car or other piece of infrastructure, such as a traffic signal, is first captured to start the video stream processing process.

A continuous stream from the live video stream needs to be handled. Preprocessing of frames to enhance the image quality and get the image ready for analysis, pre-processing is required for each frame of the video stream. Resizing, Noise Reduction, Contrast Adjustment, and Image Stabilization are some of the pre-processing techniques used to account for camera shake

**Object localization and detection**:

Using object detection methods, potholes are often located in each frame. CNNs and other deep learning-based models, as well as more sophisticated architectures like You Only Look Once (YOLO), can be used for this. These models can recognize potholes from photographs of them.

**Frame-by-frame analysis**:

Each frame of the video stream is examined. The trained model examines each frame to pinpoint the potholes and where they are in the image. When potholes are discovered in the image, this step typically entails drawing a bounding box around them which is not trained for this particular model but will be considered as one of the future implementations.

**Tracking and monitoring:**

To ascertain the potholes' persistence and gauge their severity in real-time detection, it is frequently essential to observe the potholes over a number of frames. Real-time pothole warnings or notifications can be issued to warn drivers or notify maintenance crews to take prompt action. Pothole locations can be displayed using visualization tools on maps or in video overlays. Performance improvement Firstly,you must enhance your detection and tracking algorithm to lessen computing burden and delay in order to obtain real-time data. To guarantee that your video frames are analysed quickly and correctly, you can employ methods like model quantization and hardware acceleration (such as a GPU). Potholes are continuously being monitored by a system that is operational around-the-clock. This implies that each frame is evaluated as soon as it appears on the camera feed, allowing for the detection and repair of potholes as well as the implementation of safety measures.

Real-time pothole detection using live video feeds is a challenging but rewarding application that can increase road maintenance and safety. This system recognizes and responds to road dangers in real-time by fusing the strengths of Computer Vision, Deep Learning, and Real-time processing.

**2.7 Experimental Results**

**Input Image 1:**

****

**Fig 6. Test\_Dataset Image(With Potholes)**

**Model’s Prediction:**

****

**Input Image 2:**

** Fig 7. Test\_Dataset Image(Without Potholes)**

**Model’s Prediction:**

****

**3. Future Work**

There are a number of promising directions for future research and advancement in real-time pothole detection and associated technologies, some of them are

1**.Enhancements to Accuracy and Reliability**: Future studies might concentrate on increasing the pothole detecting systems' precision and dependability. In order to do this, deep learning models may need to be improved. The variety of training data may also need to be increased.

2**.Multi-Sensor Integration** - Integrating video feeds with information from additional sensors, such as radar or LiDAR (Light Detection and Ranging), can give a more thorough view of the state of the roads. Potholes and other road dangers may be more easily detected with the use of multisensory fusion.

3. **Advanced tracking and forecasting**: To evaluate the severity of potholes and forecast their future circumstances, more complex tracking and prediction systems can be developed.We can also add bounding boxes around the potholes and can calculate their area’s which can then be used to measure their characteristics.This can help in setting maintenance priorities.

4. **Real-time notifications and alerts**:

- The real-time alerting system has to be improved. Road safety can be improved by integrating GPS and vehicle communication systems to offer automated notifications to drivers when they approach a detected pothole.

5. **Crowdsourced data collection**: Using information gathered from vehicles with pothole detecting devices, a dynamic and comprehensive database of road conditions may be produced. Road authorities can use this information to better plan for road repair.

6. **Autonomous Vehicle Integration**:

- As autonomous vehicles become more prevalent, integrating pothole detection systems into these vehicles can assist in avoiding road hazards and optimizing route planning.

7. **Edge Computing and Low Latency**:

- Reducing latency in pothole detection systems is crucial for real-time response. Utilizing edge computing and faster processing hardware can minimize delays in identifying and reporting potholes.

8. **Data Privacy and Security**:

- As these systems collect and transmit video data, ensuring data privacy and security will be essential. Robust encryption and anonymization techniques should be a focus of future developments.

9. **Environmental Impact**:

- Consideration of the environmental impact of pothole detection systems, such as the energy consumption of hardware, can be an area of improvement.

**4. Conclusion**

In conclusion, the field of real-time pothole detection using live video feeds and advanced technologies holds immense promise for enhancing road safety, reducing maintenance costs, and improving the overall quality of transportation infrastructure. This technology, driven by computer vision, AI, and IoT advancements, has the potential to revolutionize how we identify and address road hazards.

Throughout this discussion, we have explored key aspects of real-time pothole detection, including the methodologies involved in frame-by-frame video processing and the evaluation metrics used to assess the performance of detection models. Additionally, we've touched on future directions and challenges that the field is likely to encounter.

As we move forward, it's essential to emphasize the importance of continued research, innovation, and collaboration among academia, industry, and government agencies. Addressing challenges related to accuracy, multi-sensor integration, real-time alerts, and privacy will be critical to the success and widespread adoption of pothole detection systems.

**5. References**

* Wikipedia, “Pothole,” 2020. [Online]. Available: HTTP: //en .wikipedia.org/wiki/Pothole
* L. Tai, G. Paolo, and M. Liu, “Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017, pp. 31-36.
* R. Karthika, and L. Parameswaran, “An automated vision-based algorithm for out of context detection in images,” Int. J. Signal and Imaging Sys. Engg., vol. 11(1), pp. 1-8, 2018.
* Y. Sudhir, V. Girish, and C.V. Jawahar, “City-Scale Road Audit System using Deep Learning,” pp. 635- 640, 2019, 10.1109/IROS.2018.8594363.
* Kim, Eunji; Lelkes, Yphtach; McCrain, Joshua (August 9, 2022). ["Measuring dynamic media bias"](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9371639). *Proceedings of the National Academy of Sciences*. **119** (32): e2202197119. [doi](https://en.wikipedia.org/wiki/Doi_(identifier)):[10.1073/pnas.2202197119](https://doi.org/10.1073%2Fpnas.2202197119). [ISSN](https://en.wikipedia.org/wiki/ISSN_(identifier)) [0027-8424](https://www.worldcat.org/issn/0027-8424). [PMC](https://en.wikipedia.org/wiki/PMC_(identifier)) [9371639](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9371639). [PMID](https://en.wikipedia.org/wiki/PMID_(identifier)) [35914125](https://pubmed.ncbi.nlm.nih.gov/35914125).
* H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, “Deep Residual Learning for Image Recognition,”, pp. 770-778, 2016, 10.1109/CVPR.2016.90.
* Dataset:- https://www.kaggle.com/datasets/atulyakumar98/pothole-detection-dataset
* Bauder, David (August 26, 2022). ["CNN management intent on changing perception of thenetwork"](https://web.archive.org/web/20220907022134/https://abcnews.go.com/Business/wireStory/cnn-management-intent-changing-perception-network-88892179). [*ABC News*](https://en.wikipedia.org/wiki/ABC_News). Archived from [the original](https://abcnews.go.com/Business/wireStory/cnn-management-intent-changing-perception-network-88892179) on September 7, 2022. Retrieved September 5, 2022.
* K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask r-CNN,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961-2969
* Robert McNeel & Associates, “Rhinoceros,” 2020. [Online]. Available: <https://www.rhino3d.com>
* Y. Sudhir, V. Girish, and C.V. Jawahar, “City-Scale Road Audit System using Deep Learning,” pp. 635- 640, 2019, 10.1109/IROS.2018.8594363.
* S. Ren, K. He, R. Girshick, and J. Sun, “Faster RCNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol.39(6), pp. 11371149, June 2017.DOI:https://doi.org/10.1109/TPAMI.2016.2577 03
* Bauder, David (August 26, 2022). ["CNN management intent on changing perception of the network"](https://web.archive.org/web/20220907022134/https://abcnews.go.com/Business/wireStory/cnn-management-intent-changing-perception-network-88892179). [*ABC News*](https://en.wikipedia.org/wiki/ABC_News). Archived from [the original](https://abcnews.go.com/Business/wireStory/cnn-management-intent-changing-perception-network-88892179) on September 7, 2022. Retrieved September 5, 2022.