



Railway Track Crack Detection System

Akhil L¹, Divya Manjari P², Anil Sai K³, Raja Sravan Kumar L⁴

^{1,2,3} Students, and ⁴ Faculty

Dept. of Information Technology, Velagapudi
Ramakrishna Siddhartha Engineering
College, Vijayawada, India.
akhillolugu44@gmail.com

Abstract: Discover how railway track maintenance will develop in the future with our ground-breaking invention, the Railway Track Crack Detection System. Envision traversing the immense 92,081 km (57,216 mi) of track and 7,216 stations spanning 66,687 km (41,437 km) of the Indian railway network. As a result of this network's immense size, manual inspection is a laborious and time-consuming process that requires a sizable human workforce. But have no fear—our article offers a revolutionary fix! You may stop worrying about crack detection since our technology not only locates defective tracks accurately, but it also promptly notifies local train stations via a short messaging app. Envision the inconveniences avoided and the lives saved!.

Keywords: Convolutional Neural Networks, Deep Learning, Derailments, Faulty Tracks

1. INTRODUCTION:

In order to improve efficiency and lessen the need for human labor, an automated railway track inspection system is presented in this study to replace manual inspection procedures. The system continually records and analyzes real-time data along the railway lines through the use of a range of sensors like high-resolution cameras. By utilizing computer vision and machine learning techniques, the system allows for the accurate identification and categorization of track problems, facilitating prompt intervention.

India has the fourth-largest railway network in the world, and it is expanding to accommodate the demands of its expanding economy. It hasn't, however, yet attained international reliability and security criteria. Derailments have been shown to be mostly caused by fractures in the rail, accounting for 60% of rail accidents. According to recent data, rail breaks either from natural causes or antisocial factors account for 90% of these derailments. Sadly, there aren't any low-cost automated options accessible for testing. A complex yet reasonably priced electronic gadget that is intended to transform railway fracture detection is at the core of an inventive track detecting system. With the help of vital components including a Wi-Fi module, cameras, GPS, and a micro-controller (Raspberry pie 4), this state-of-the-art technology creates a crack detecting assembly that might revolutionize railway safety. By substituting the present human inspection process, the suggested setup would facilitate the inspection and maintenance of railway lines and aid in their efficient monitoring. The vehicle's design and associated software are fairly straightforward, making them easy to integrate into the current system. Envision a patrolling car moving smoothly down the rails at regular intervals, looking attentively for any fractures. With this cutting-edge detecting technology installed, the car patrols the railway network like a watchful escort, quickly spotting problems.

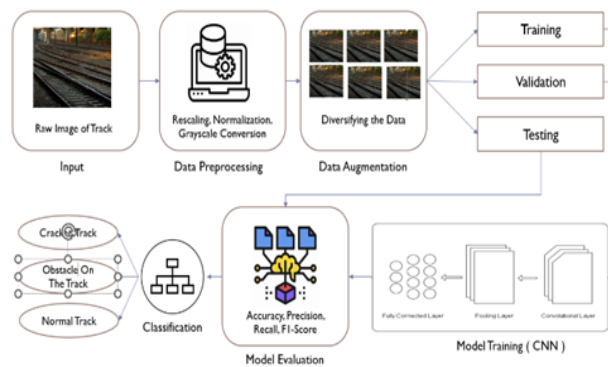
2. REVIEW OF LITERATURE:

Effective railway line inspection is crucial for predictive maintenance. Manual inspection by humans is time-consuming and susceptible to errors. To enhance efficiency, automated systems have been developed. Non-destructive evaluation techniques play a key role in rail track inspection. The literature is categorized based on electromagnetic, guided, vision, IoT, and acoustic approaches [1]. The proposed autonomous testing train attempts to improve railway safety by detecting cracks in rails using magnetic flux leakage (MFL) detection. Often employed in many different industries, this non-destructive technique guarantees the detection of even the smallest fissures. Hall sensors are used in the system to improve accuracy, and SMS real-time communication informs officials of the precise position of track defects so that timely action may be taken. Because the testing train is autonomous, it can handle irregular maintenance problems that arise from manual monitoring and offers a reliable and comprehensive inspection of the whole railway track. All things considered, this creative method may help keep railroad systems safer overall, avert accidents, and save lives[2]. The primary objectives of the proposed study is to create and implement a system that will efficiently detect, locate, and notify the relevant authorities of any fractures or other metallic problems. Two laser transmitters, a Raspberry Pi 3 model, a GPS (Global Positioning System) module, a GSM module, and an image processing-based track detection framework are all included in the suggested robot. A laser transmitter and receiver will detect any track cracks. The railway line damage is taken by the camera, and the resulting image is then used for analysis. The Raspberry Pi 3 receives the signal from GPS, and the GSM modules then transmit an alarm message containing the region's longitudinal coordinates to the authorized contact that has already been registered in the robot[3]. Railway track cracks stand as a major contributor to accidents. Traditional manual inspection, being time-consuming and error-prone, calls for innovative crack detection technologies. This project leverages sensors and a GPS tracking system, with Arduino microcontrollers overseeing device operations. IR sensors gather data for subsequent computational analysis, and the collected data is stored in the cloud for comprehensive examination[4]. Efficient detection of cracks and obstacles on railway tracks is imperative. Manual analysis proves time-consuming and ineffective. The proposed system introduces a solar-powered moving vehicle for detection, integrating sensors and GPS/GSM modules. Detected areas are promptly marked, and the location information is transmitted to the relevant authorities. Comparative analysis with conventional methods validates the system's efficiency [5]. The detection of cracks on railway tracks is achieved through computer vision, employing deep learning for object and fracture detection. Image pre-processing, along with the application of Gabor filters, enhances image quality. A deep learning neural network is utilized to differentiate between cracked and non-cracked track images, recognizing objects present on the tracks. The architecture includes convolutional, RELU, pooling, and fully connected normalized layers in a CNN framework featuring shared weights and local connections. The training dataset comprises photographs of both cracked and uncracked tracks[6]. The envisioned robot serves the purposes of crack detection, collision avoidance, and energy conservation. It incorporates a range of sensors, including IR, ultrasonic, and touch sensors. The sensor data is regularly updated on a designated IoT website. The development process involves the use of Arduino IDE and Proteus software. The robot contributes significantly to the maintenance and monitoring of railway track conditions. In the event of a potential collision, it promptly alerts the train driver via SMS. Other studies have explored the utilization of GSM technology and wireless sensors in a similar context [7]. To ensure effective railway track maintenance, surface analysis for faults is essential. A computer vision system is employed for inspecting defects on the rail surface, incorporating algorithms for noise reduction, filtering, and thresholding. The process includes segmentation and feature extraction to analyse images of defects. While currently implemented on computers, there is potential for enhancement through embedded systems. Adaptive thresholding and feature matching are utilized for precise flaw extraction. The experimental results and benchmarks are thoroughly presented in the results section[8].

3. ARCHITECTURE:

This section describes the proposed methodology, the machine learning methods, and the dataset acquisition procedure that were applied to the categorization. Figure 1 shows how the suggested prototype would function. It could be used to detect a problematic track and send position data via a Telegram bot service created with Bot-father, a useful Telegram page for building bots. A Raspberry Pi micro-controller with an HD camera is utilized to take pictures that are employed to categorize if a track is defective or not, or whether anything has been placed on the track. The Raspberry Pi (we have used version 4) classifies images based on deep learning which is trained model using CNN algorithm. Our methodology comprises an all-encompassing strategy with the goal of creating a strong classification model. It is based on the collecting and processing of data. Data augmentation, investigating several CNN architectures, training of models, and evaluation were the next phases. The general project procedure is shown in Fig. 1.

Fig. 1. Architecture diagram for the proposed model



4. FLOWCHART:

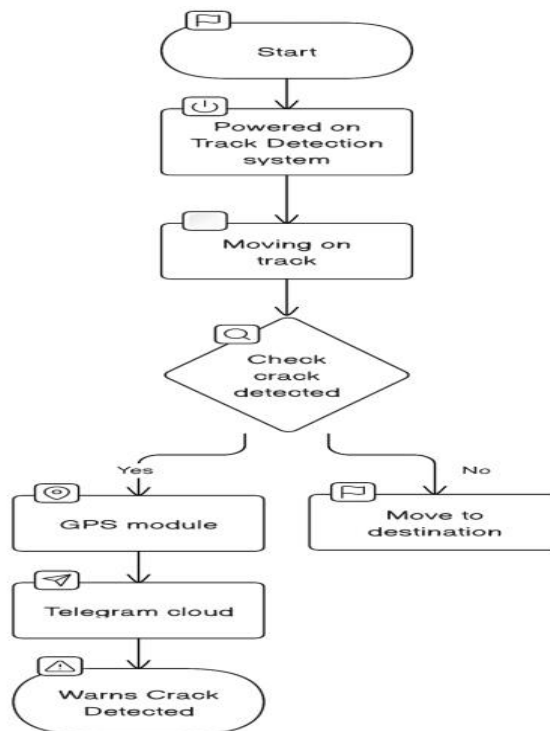


Fig. 2. Flowchart diagram for Railway Track Crack Detection

A.Data Acquisition

Our dataset, sourced from Kaggle, comprises 5154 instances of 2D railway track samples capturing the axial view of the tracks. Each image maintains a standardized resolution of 170×256 pixels, utilizing 8 bits per pixel and 96 dots per inch (dpi). Within this dataset, we categorized 1124 instances of track faults, 2590 instances of objects being placed on tracks, and 1440 instances of the normal tracks. This Kaggle-sourced dataset forms the bedrock of our exploration into deep learning for classifying the different types of railway track faults.

B.Data Preprocessing`

As we navigate the details of our dataset, a crucial step is the data pre-processing phase. In this transformative process, we performed a series of operations to tune and improve our dataset, thereby laying the groundwork for optimal model training.

1)Image Resizing: In initial stage of data processing, we prioritized standardization by resizing all images from their original dimensions of 170×256 to a more manageable 128×128. This adjustment ensures uniformity in the input size, thereby promoting computational efficiency without compromising the vital image information. This thoughtful resizing contributed to an optimized dataset for subsequent model development.

2) Grayscale Conversion: To improve computational efficiency, we switched from RGB to grayscale during the image conversion process. We streamlined the dataset by reducing the number of color channels from three to one, focusing on essential image features while minimizing computational complexity. This simplification optimizes the learning process of the model, ensuring that computing resources are used efficiently without sacrificing critical information.

3)Normalization: A crucial step in data preprocessing involves MinMax normalization, in which we constrain the range of pixel values to fall between 0 and 1. This normalization technique is instrumental in promoting stable and consistent learning across datasets, ensuring that the model interprets the pixel values in a standardized manner. By limiting the pixel value range, we facilitated effective model training and generalization, contributing to the development of a robust and accurate classification model.

C.Data Augmentation

Data augmentation is a pivotal technique employed in machine learning, serving the fundamental purpose of enhancing the diversity and size of a dataset. Through a set of operations, such as random flipping and translation, data

augmentation introduces variations to existing data, effectively expanding the range of perspectives that a model is exposed to during training. This improves the model's generalization ability, addresses class imbalances, optimizes model resilience, and increases the dataset size.

1)Random Horizontal Flipping: This technique enriches our dataset by introducing variability through the horizontal flipping of images. This expands the dataset's perspectives and enhances the ability of our model to discern diverse features.

2) Random Translation: A controlled augmentation approach involving subtle shifts, both positive and negative, within a range of 10%. This translation introduces additional perspectives to images by contributing to a more comprehensive and diversified dataset.To ensure a balanced representation across classes, we generated new images, resulting in a total of 7800 images (2600 for each of the three classes). This approach fortifies our model, paving the way for improved generalization.

D.Algorithms

We explored various CNN architectures, such as LeNet-5, VGG 16, and AlexNet, to advance the classification of Alzheimer's disease. On the other hand, our specially designed CNN model stood out. Our model achieved an impressive accuracy of approximately 96%. Our model is distinguished by its intricately designed structure, which includes convolutional layers, an Exponential Linear Unit (ELU) as the activation function, and batch normalization for accurate classification. The performance metrics, including precision, recall, and f1-score, reached approximately 95%, showing significant parameters for model evaluation.

To optimize the architecture of our model, we strategically placed convolutional layers followed by ELU activation functions, thereby introducing nonlinearity to the system. In addition, batch normalization was employed for stable training. To enhance training efficiency, we implemented an early stopping mechanism that closely monitored the validation error for each epoch. If the validation error failed to decrease for a predetermined number of epochs, the training process was halted. This ensures an optimal performance and reinforces the robustness of the proposed approach.

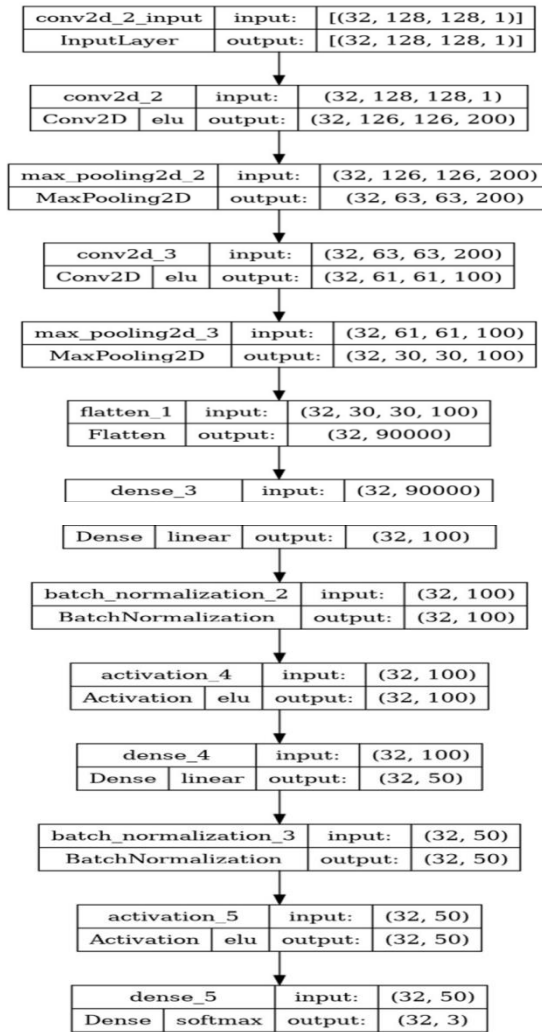


Fig. 3. Proposed Model Architecture

Fig 3 illustrates the architecture of the proposed model. The architecture of our model is structured sequentially, beginning with a convolutional layer employing 200 filters with a 3×3 kernel size and utilizing an Exponential Linear Unit (ELU) activation function. Subsequently, max-pooling with a 2×2 pool size is employed to reduce the spatial dimensions. The subsequent convolutional layer introduced 100 filters of size 3×3 with ELU activation followed by another max-pooling operation. The flattening

layer serves the purpose of transforms the output from the convolutional layers into a format suitable for the dense layers. Moving on to the dense layers, the model incorporated a dense layer with approximately 100 neurons and an ELU as the activation function. Batch normalization was applied to enhance the training stability. Another dense layer features approximately 50 neurons, batch normalization, and ELU activation. The final layer, also known as the output layer, consisted of a dense layer with softmax as the activation function. This layer generates probabilities for each class, thereby providing a prediction for the model. The He_Normal kernel initializer is applied to the convolutional and dense layers. It is specifically designed for deep neural networks, and it initializes the weights of neurons in a manner that helps address the vanishing gradient problem. The scale of the weights is adjusted based on the number of input and output units in the layer. This helps to maintain a balanced scale of activation throughout the network, contributing to smoother and more stable training. We employed Exponential Scheduling, a powerful technique for dynamically adjusting the learning rate during training. This technique proved to be beneficial in optimizing the ability of the model to learn intricate features from the dataset. The primary objective was to enhance the convergence efficiency of the CNN model during training.

E. Equations :

1) *Softmax Function*: The softmax function shown in Eq. (1) is a mathematical function that takes the input as a vector of real numbers and normalizes it into a probability distribution. It is commonly used in the output layer of neural networks for multiclass classification problems. The softmax function transforms an N-dimensional vector (z_1, z_2, \dots, z_n) into a probability distribution (p_1, p_2, \dots, p_n) as shown in (1)

$$\text{Softmax}(z)_i = \frac{e^{z_i}}{\sum_{j=1}^n e^{z_j}} \quad (1)$$

Where, z_i is the i th element of input vector z .

2) *Categorical Cross Entropy Loss*: The categorical cross-entropy (2) is a loss function commonly used in machine learning, particularly in the context of neural network training for multiclass classification problems. It measures the dissimilarity between the true distribution of target classes and the predicted probability distribution generated by the model.

$$\text{Loss} = - \sum_{i=1}^n \sum_{j=1}^k (y_{ij} \log \hat{y}_{ij}) \quad (2)$$

Where, y_{ij} is the actual value and \hat{y}_{ij} is the predicted value of j th instance of i th class respectively.

3) *Exponential Linear Unit (ELU)*: The Exponential Linear Unit (ELU) shown in Eq. (3) is an activation function commonly employed in neural networks. It was designed to address the vanishing gradient problem, which can impede the training of deep neural networks. ELU introduces smooth nonlinearity to the network by allowing negative values, unlike traditional activation functions such as rectified linear units (ReLUs), which discard negative values. It has been used in convolutional and fully connected layers, except for the output layer in the proposed model architecture.

$$\text{ELU}(x) = \begin{cases} x, & x \geq 0 \\ \alpha(e^x - 1), & x < 0 \end{cases} \quad (3)$$

Where, x is the input vector and α is leakage hyperparameter.

5. RESULTS :

The following figure illustrates the actual model of proposed project

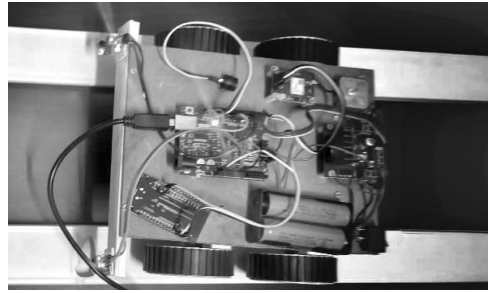


Fig 4 : Prototype of proposed project

Fig. 5 shows the confusion matrix for the proposed architecture. The confusion matrix categorizes predictions into true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each class. The three classes were AD, Cognitive Impairment (CI), and normal. We compared various convolutional neural network (CNN) architectures, including LeNet 5, VGG 16, AlexNet, and the proposed model, and evaluated their classification performance. The evaluation metrics employed for each model included accuracy, precision, recall, and f1-score. Because it is a multiclass classification, there are individual metrics, such as precision, recall, and f1-score, for each of these classes.

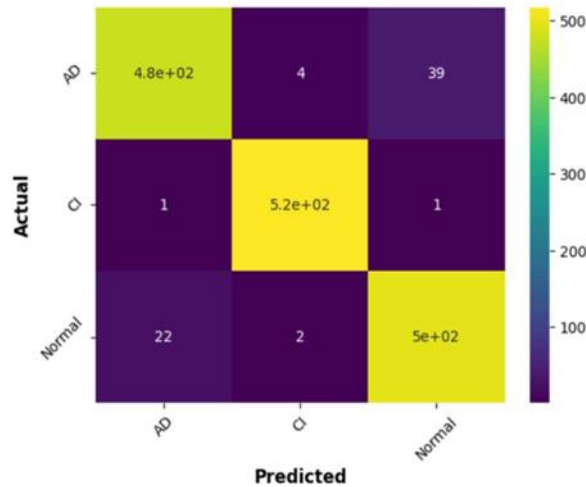


Fig 5 : Confusion matrix of the proposed system

We tabulated the results of these metrics in Table 1, which offers a comprehensive comparison of the performance of the models. Table 2 shows a comparison of the different architectures along with our proposed model in terms of their evaluation metrics.

TABLE I. EVALUATION METRICS FOR PROPOSED MODEL

	Evaluation Metrics		
	Precision	Recall	F1-Score
Faulty Tracks	0.954	0.917	0.935
Objects on the track	0.989	0.996	0.992
Normal	0.925	0.954	0.939
Macro Average	0.957	0.955	0.956
Weighted Average	0.957	0.958	0.957

TABLE II. COMPARISON OF DIFFERENT ARCHITECTURES

Model	Evaluation Metrics			
	Accuracy	Precision	Recall	F1-Score
LeNet 5	0.856	0.844	0.849	0.846
VGG 16	0.884	0.872	0.877	0.874
AlexNet	0.928	0.912	0.924	0.918
Proposed Model	0.956	0.957	0.958	0.957

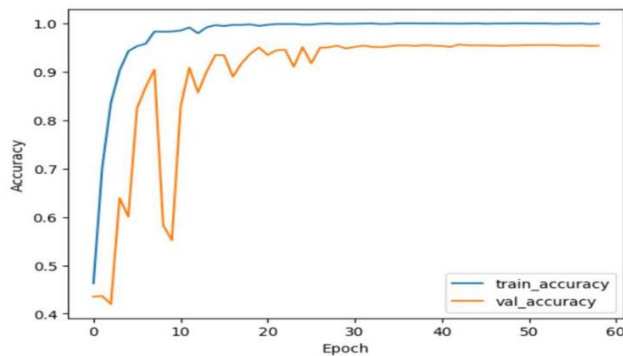


Fig. 6 . Accuracy curves for the proposed model

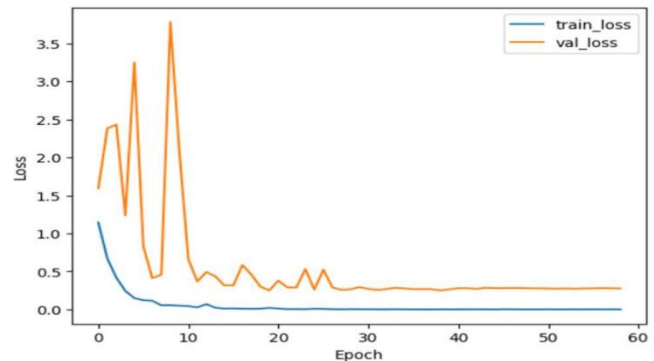


Fig. 7. Loss curves for the proposed model

When a crack is being detected the proposed model will send the co-ordinates of the crack to nearest Railway Stations via a Telegram Bot.They will take further action. The system contains a buzzer which will alert people around the faulty track so that they can stop incoming trains.

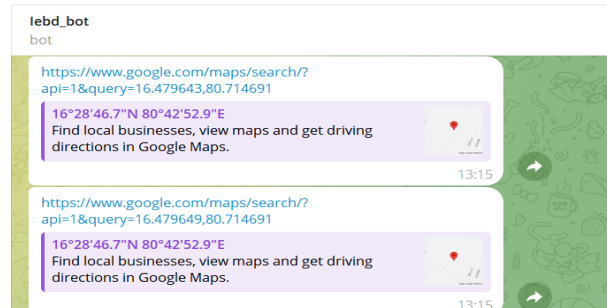


Fig. 8 . Location Co-ordinates Detected By The Proposed System via Telegram Bot Service

6. CONCLUSION

Cracks in railway tracks have historically been a primary catalyst for derailments, underscoring the critical need for proactive solutions. Our project focuses on addressing this issue by introducing an efficient, cost-effective, and autonomous system capable of detecting faulty rail tracks without human intervention. This automated approach not only minimizes the risk of accidents but also transcends traditional detection methods in terms of cost-effectiveness, power efficiency, and analysis time. Our Proposed system has 95.64 % accuracy in detecting the faulty tracks which we think is really helpful in the Indian Railways.

7. References:

- [1] H. U. R. Siddiqui, A. A. Saleem, M. A. Raza, K. Zafar, K. Munir and S. Dudley, "IoT Based Railway Track Faults Detection and Localization Using Acoustic Analysis," in *IEEE Access*, vol. 10, pp. 106520-106533, 2022, doi: 10.1109/ACCESS.2022.3210326.
- [2] S. N, G. M. Iype, K. C. P. M. Sharon and S. Subhash, "Rail Track Defect Detection using Enhanced Method of Magnetic Flux Leakage Signal," 2021 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C), Bangalore, India, 2021, pp. 277-280, doi: 10.1109/ICDI3C53598.2021.00062.
- [3] J. N. L. A, L. Narayanan C, K. M. S, R. S and K. R, "Railway Track Crack and Key Detection Robot using IoT," 2022 International Conference on Edge Computing and Applications (ICECAA), Tamilnadu, India, 2022, pp. 623-628, doi: 10.1109/ICECAA55415.2022.9936435.
- [4] Sana P, Fathima, Amsu Krishnan, and Ahammed Muneer KV. "Design and Implementation of Solar Powered Moving Vehicle for Railway Track-Crack and Obstacle Detection." Amsu and KV, Ahammed Muneer and M, Nivedh, Design and Implementation of Solar Powered Moving Vehicle for Railway Track-Crack and Obstacle Detection (April 11, 2023).
- [5] Rangineni, Sandeep, and Divya Marupaka. "Analysis Of Data Engineering For Fraud Detection Using Machine Learning And Artificial Intelligence Technologies." *International Research Journal of Modernization in Engineering Technology and Science* 5.7 (2023): 2137-2146.
- [6] Kavitha, V., and M. MAHESHWARI. "Iot Based Crack Detecting, Obstacle Detecting and Energy Minimizing Robot in Railway Track." *International Journal of Innovations in Engineering Research and Technology* 6.6: 1-5.
- [7] Thirumal, P. C., et al. "RAILWAY TRACK CRACK DETECTION SYSTEM."
- [8] N. Patil, D. Shahare, S. Hanwate, P. Bagde, K. Kamble and M. Titre, "Designing of Improved Monitoring System for Crack Detection on Railway Tracks," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 2021, pp. 1514-1517, doi: 10.1109/ICICV50876.2021.9388429.
- [9] "Railway Track Crack Detector," Karpagam V a, Naveeshwaran Mb*, Ajay Kumar M c, Rajashree Kd, *International Journal of Research Publishing and Reviews*, Vol 3, number 6, pages 2716-2719, June 2022
- [10] S. Sree Nandha, V. P. Athish and D. Rajeswari, "Train Track Crack Prediction Using CNN with LeNet - 5 Architecture," 2022 International Conference on Futuristic Technologies (INCOFT), Belgaum, India, 2022, pp. 1-5, doi: 10.1109/INCOFT55651.2022.10094528.
- [11] S. Rakshit, S. BS and E. F. S. S, "Railway Track Fault Detection using Deep Neural Networks," 2022 IEEE 6th Conference on Information and Communication Technology (CICT), Gwalior, India, 2022, pp. 1-5, doi: 10.1109/CICT56698.2022.9997883.
- [12] W. Yu, M. Rong, D. Min, Z. Ying and Z. Guang, "Research on rail surface crack identification and detection based on deep learning and image processing," 2023 2nd International Conference on 3D Immersion, Interaction and Multi-sensory Experiences (ICDIIME), Madrid, Spain, 2023, pp. 187-190, doi: 10.1109/ICDIIME59043.2023.00040.

