Railway Safety Image Classification Model

Introduction:

Railway safety is a critical concern, especially during the arrivals and departures of trains at stations. The safety of both passengers awaiting transportation and those aboard the train is paramount. In regions like India, where unauthorized track crossings and standing perilously close to platform edges are commonplace, ensuring safety becomes a pressing challenge. These risky behaviors pose threats not only to individuals at the station but also to the integrity of the trains themselves.

In response to these safety challenges, our project endeavors to harness the potential of cutting-edge technology—specifically, leveraging deep learning methodologies. By employing Convolutional Neural Network (CNN) models, our primary goal is to develop a robust system capable of detecting and identifying various objects and elements within and around railway platforms. This initiative aims to enhance railway safety by proactively identifying potential hazards, thus mitigating risks associated with unsafe practices, ultimately safeguarding both passengers and the railway infrastructure.

Our model's foundation lies in the meticulous identification of nine distinct objects prevalent in railway station environments. With a total dataset comprising 361 images, of which 284 images are allocated for training and 77 for testing, we aim to comprehensively cover diverse scenarios encountered at railway platforms. Most of the dataset consists of primary data, meticulously collected through firsthand image capture. Additionally, to augment our dataset, supplementary images have been sourced from secondary sources like Google Images and Pinterest, ensuring their compliance with free-to-use permissions. This amalgamation of primary and secondary data enriches our model's training dataset, fostering a robust and inclusive approach to object detection and safety enhancement within railway environments.

Problem Statement:   
Ensuring railway safety stands as a pivotal concern, particularly during train arrivals at stations. The safety of both passengers aboard the train and individuals waiting at the station is crucial. This concern gains paramount significance in the context of India due to prevalent practices like unauthorized track crossings and standing too close to platform edges. These actions pose severe risks, potentially causing harm to station-goers, train passengers, and even the trains themselves.

To address this challenge, our project aims to harness the power of Convolutional Neural Network (CNN) models. By utilizing images captured at railway stations, our objective is to develop a sophisticated system capable of detecting and recognizing various objects and elements present on the station premises. Through this approach, we aim to enhance railway safety by identifying potential hazards, thus minimizing risks associated with unsafe practices and ultimately ensuring the well-being of both passengers and railway infrastructure.

Literature Review:

# **A Deep Learning Approach Towards Railway Safety Risk Assessment**

[Hamad Alawad](https://ieeexplore.ieee.org/author/37087229343); [Sakdirat Kaewunruen](https://ieeexplore.ieee.org/author/37085874601); [Min An](https://ieeexplore.ieee.org/author/37085879010)

The study proposed leveraging deep learning, particularly Convolutional Neural Networks (CNN), to advance risk management in the railway sector. Deep learning showcased benefits in real-time risk avoidance, integrating maintenance, security, and traffic systems, automating safety processes, and improving operational efficiency. However, challenges in acquiring labeled datasets hindered accurate model training for identifying station risks, despite automated risk detection aiding in timely maintenance. The CNN method exhibited versatility beyond fall detection, identifying various risks like overcrowding, offering cost-effective real-time monitoring suitable for diverse conditions. It demonstrated high accuracy in detecting risky behaviors, enabling interventions to mitigate accidents. Further specialized algorithms could enhance accuracy and response times. While leveraging CCTV systems reduced long-term costs and enabled predictive maintenance, challenges persisted in data availability and quality. Overall, investing in AI remained crucial for ensuring railway safety for both staff and the public.

# **A Review of Deep Learning Applications for Railway Safety**

[**Kyuetaek Oh,**](https://sciprofiles.com/profile/1095490?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Mintaek Yoo,**](https://sciprofiles.com/profile/342320?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Nayoung Jin,**](https://sciprofiles.com/profile/author/TWVRdUgzR25MbE1UeWZaYUhpRkRqaC8vU2pMdDJOVERXKzBXcEltVmxyMD0=?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Jisu Ko,**](https://sciprofiles.com/profile/2494082?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Jeonguk Seo,**](https://sciprofiles.com/profile/author/YTVMODF6MkJJUGRtZUcyUmx5cnBXYVJidWR0VVNnNXJvbERyUmZTejdSVT0=?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Hyojin Joo,**](https://sciprofiles.com/profile/author/YUl2L09DRGdXRUt3NVdSM3JGenVtc0tMWUJuVjNYanBValpYdVVLSlZlQT0=?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Minsam Ko**](https://sciprofiles.com/profile/787715?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)

The study highlights the extensive applications of artificial intelligence (AI) in railway safety, spanning infrastructure, operations, and train-related aspects. It emphasizes AI's potential to enhance conventional safety methods and automate inspection procedures reliant on visual analysis or expert knowledge. AI models predominantly employed image/video data for defect detection, while train vibrations were analyzed using LSTM-based models.

However, challenges persist in optimizing AI for railway safety. Performance enhancement and generalization stand as crucial issues. Improving model accuracy despite limited labeled data and reducing processing time for real-time applications are focal points. Moreover, the study underscores the need for diverse data sources and model validation in real-world railway settings.

Addressing data scarcity, processing efficiency, and embracing varied data sources like LiDAR and acoustic signals could bolster AI's efficacy. Model frameworks focusing on defect detection should aim for broader applicability, necessitating validation with real train data for enhanced practicality and accuracy.

# **Real-Time Hybrid Deep Learning-Based Train Running Safety Prediction Framework of Railway Vehicle**

[**Hyunsoo Lee,**](https://sciprofiles.com/profile/459003?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Seok-Youn Han,**](https://sciprofiles.com/profile/904071?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)[**Keejun Park,**](https://sciprofiles.com/profile/author/ejRlbmdnSENmTkM3UmtRVXRXOGNiQnFBak1DYmI3NmRzYUNUaDEvQk1xcz0=?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name) **Hoyoung Lee,** [**Taesoo Kwon**](https://sciprofiles.com/profile/author/eWFtNk9xZlRWd3VVNjY3UzJtTThQQURGeXdFMWdHNHJxODNhb1plSXpZND0=?utm_source=mdpi.com&utm_medium=website&utm_campaign=avatar_name)

This study introduces an innovative hybrid deep learning framework aimed at predicting train running safety, a crucial aspect often overlooked in train safety assessment. By considering five key output factors related to train safety and analyzing twelve vibration-based signals across six railway conditions, this framework utilizes a hybrid architecture incorporating recurrent data for enhanced prediction accuracy.

Combining elements of a general Deep Neural Network (DNN) and a Recurrent Neural Network (RNN), this framework exhibits superior performance compared to LSTM and DNN architectures in predicting real-time train safety. The proposed model's potential application in advanced train control systems to mitigate derailment risks and prevent potential accidents underscores its significance for passenger and cargo safety.

Future studies could explore integrating this framework into real-time train control systems and considering additional recurrent deep learning mechanisms for further improvement. Overall, this novel hybrid deep learning approach presents a promising avenue for more accurate prediction of train running safety by incorporating crucial railway conditions and vibration signals.

Challenges Faced:

I encountered significant challenges in gathering both primary and freely available secondary data for my project.

Securing primary data, which encompasses firsthand information directly collected for the project, proved to be a notable challenge. Additionally, sourcing secondary data that was freely accessible and permissible for use posed another obstacle. These difficulties in acquiring relevant and unrestricted datasets significantly impacted the initial stages of my research.

Dataset:

Link: [Railway Safety Dataset](https://drive.google.com/drive/folders/1A4ZY51O7dCHzGpvGRYugHNDMAe0v_knM?usp=drive_link)

In our dataset, we have meticulously curated a collection of 361 high-quality images, encompassing a diverse range of scenes encountered in railway station environments. Within this dataset, we have distinctly categorized nine classes, each representing specific objects or elements commonly found in and around railway platforms. These classes encompass a comprehensive array of vital components crucial to railway safety and operational efficiency.

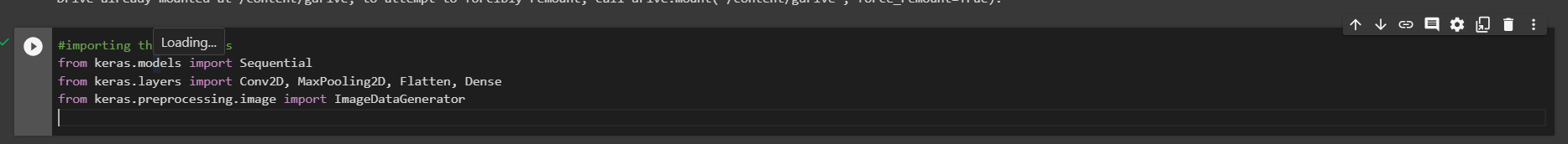
9 Classes:

1. Crowd (People on Tracks)
2. People (People on the platform)
3. Entering (Train Arriving)
4. Exiting (Train Departure)
5. Standing (Train standing on the station)
6. Faulty (Faulty tracks)
7. Name (Station sign Board)
8. Shops (Shops on the platform)
9. Track (railway tracks)

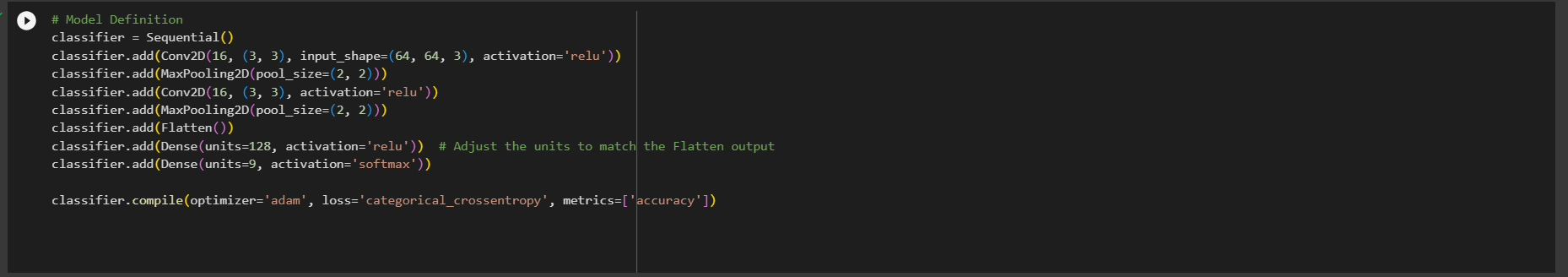
Codes:



These lines are used to mount Google Drive to your Colab notebook. The first line imports the necessary module from Colab to connect to Google Drive. The second line mounts your Google Drive in the Colab environment and sets the root path as 'gdrive/My Drive/'.



These lines import necessary modules from Keras, including the Sequential model and layers (Conv2D, MaxPooling2D, Flatten, Dense), as well as ImageDataGenerator for data augmentation.



1. Conv2D:This layer initializes the CNN with a Convolutional operation. It uses 16 filters/kernels of size 3x3 with a ReLU activation function. The input shape is set to (64, 64, 3), representing an image size of 64x64 pixels with 3 color channels (RGB).

2. MaxPooling2D: Following each Conv2D layer, MaxPooling2D is applied with a pooling size of 2x2. This operation reduces the spatial dimensions (width and height) of the input, helping to extract the most important features and reduce computation.

3. Flatten: This layer converts the 2D feature maps obtained from the convolutional layers into a 1D array, preparing the data for input into the fully connected layers.

4. Dense (Fully Connected) Layers: Two Dense layers follow the Flatten layer. The first Dense layer consists of 128 neurons with a ReLU activation function. The final Dense layer has 9 neurons (equal to the number of classes) with a softmax activation function for multi-class classification. Softmax normalizes the output into a probability distribution across the classes, making it suitable for classification tasks.

Mathematical Operations:

Convolutional Layer: In each Conv2D layer, the operation involves a set of 3x3 filters sliding over the input image, performing element-wise multiplication and addition (dot product) to produce feature maps. The ReLU activation function introduces non-linearity, allowing the model to learn complex patterns in the data.

MaxPooling: MaxPooling2D with a pool size of 2x2 reduces each spatial dimension (width and height) of the feature maps by taking the maximum value within each pooling window.

Flatten: This layer reshapes the output from the previous layers into a 1D array by flattening the feature maps.

Dense Layers: The Dense layers perform matrix multiplication of the input data with the weights and apply the activation function (ReLU for the first Dense layer and softmax for the final output layer). These layers introduce non-linear transformations to the data, enabling the model to learn complex relationships between features and classes.

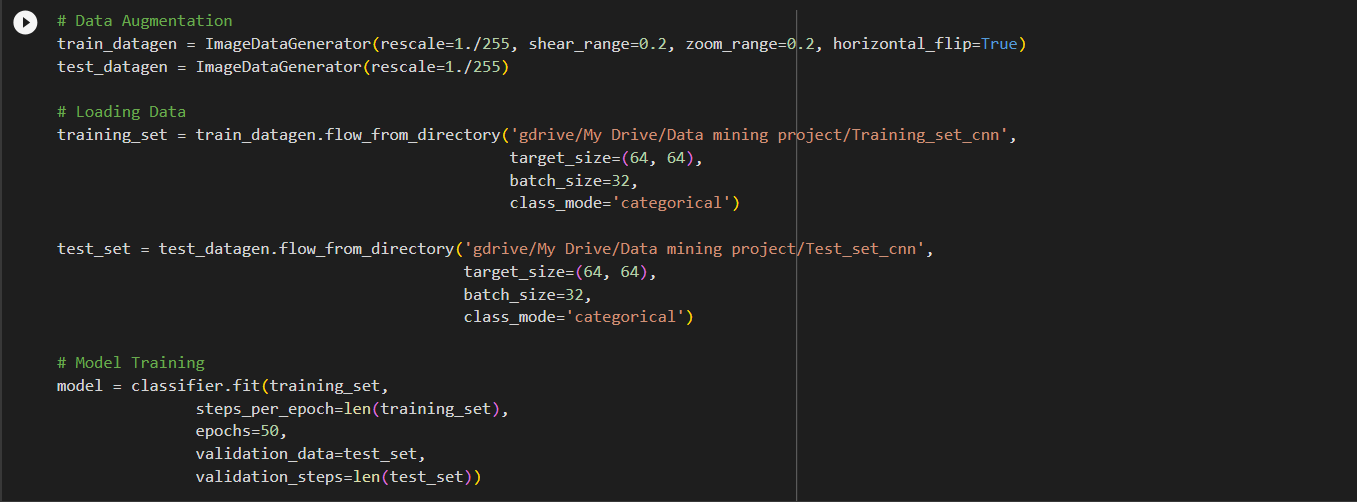
Model Compilation:

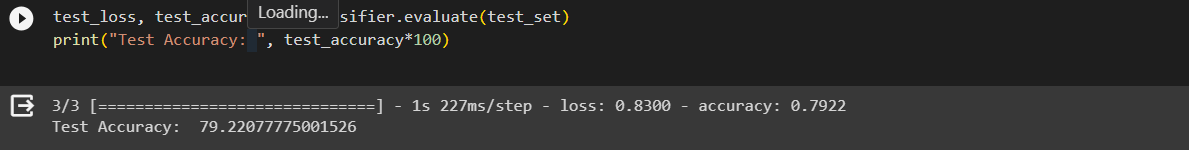
Optimizer: 'Adam' is used as the optimizer. Adam is an adaptive learning rate optimization algorithm that efficiently updates network weights based on adaptive learning rates for each parameter.

Loss Function: 'Categorical Crossentropy' is the chosen loss function for multi-class classification. It calculates the loss between predicted probabilities and the actual categorical labels.

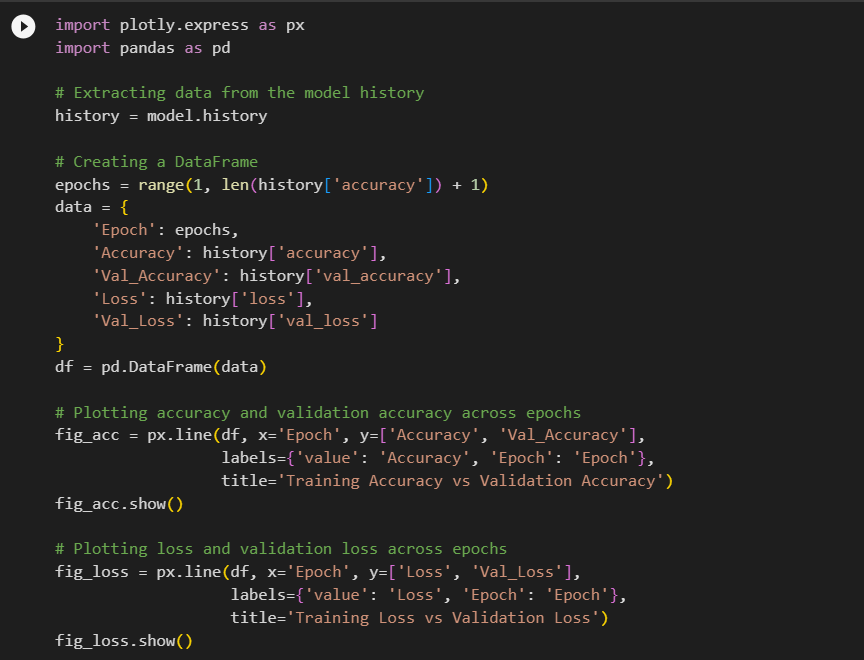
Metrics: The model's performance during training and evaluation is measured using the 'accuracy' metric, which calculates the accuracy of predictions compared to the true labels.

This CNN architecture, through its series of convolutional, pooling, and fully connected layers, learns to extract hierarchical features from input images and make predictions on the identified classes based on these learned features.





These lines set up data augmentation for the training and test datasets. `ImageDataGenerator` is used to perform real-time data augmentation and normalization. This code generates batches of augmented data from directories containing images. It sets up the training and test data generators for feeding into the model. This line trains the defined model (`classifier`) using the training data (`training\_set`) for 50 epochs while also validating it on the test set (`test\_set`). It stores the training history in the `model` variable. After running the epoch, we can see that the accuracy being found is 79%.



The following code uses Plotly Express to visualize the training/validation accuracy and loss across epochs, as previously described. This generates line plots displaying the training/validation accuracy and loss metrics over the epochs to visualize the model's performance during training.



Future Implementation:

Expanding our dataset with additional primary data and refining the model's accuracy stands as a pivotal step in further enhancing the efficacy of our approach. With a continuous influx of primary data, we aim to fortify the model's capability to accurately detect and identify objects within and around railway stations.

This advancement serves as a steppingstone toward the practical implementation of our methodology. By integrating this machine learning model into the surveillance cameras stationed at railway platforms using Internet of Things (IoT) technology, we envision a proactive monitoring system. This system would enable real-time monitoring of railway stations, facilitating the swift identification of potential hazards or risky behaviors.

By leveraging this amalgamation of cutting-edge technologies, we aspire to create a comprehensive safety framework. Such an initiative holds the promise of significantly contributing to the preservation of both passenger well-being and the integrity of railway operations. Ultimately, this innovative solution endeavors to ensure the safety and security of individuals present at railway stations while bolstering the overall safety measures for trains and station premises.

Refernces:

<https://ieeexplore.ieee.org/abstract/document/9102292/authors#full-text-header>

<https://www.mdpi.com/2075-1702/9/7/130>

<https://www.analyticsvidhya.com/blog/2018/12/guide-convolutional-neural-network-cnn/>

<https://www.analyticsvidhya.com/blog/2022/03/basics-of-cnn-in-deep-learning/>

<https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/>