Report for Traffic Signs Recognition

Names of Student:
Rehab
Sama
Raghad

8/15/2024

Problem statement

The goal is to develop a deep learning model for traffic sign recognition. The system must accurately detect and classify various traffic signs under diverse conditions, such as different lighting, weather, and angles. This model will enhance road safety by providing timely information to drivers or autonomous vehicles, ensuring compliance with traffic laws.

- **Variability in Lighting and Weather:** Images may be taken under different lighting conditions and weather situations, affecting visibility.
- **Different Angles and Distances:** Signs can appear in various orientations and distances from the camera.
- **Occlusions**: Signs may be partially obscured by objects (e.g., trees, vehicles).
- **Class Imbalance**: Some sign classes may have significantly more samples than others.

Objective

Develop a system for accurate detection and recognition of traffic signs from images to enhance road safety and support autonomous driving.

Data collection

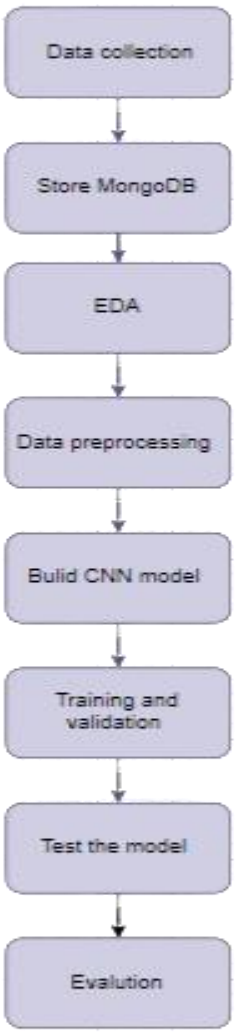
The dataset is intended for the classification of traffic signs, comprising approximately 58 distinct classes. Each class contains around 120 images, which provide a diverse set of samples for training and evaluation. The dataset also includes a labels.csv file, which contains descriptions for each traffic sign class. The goal is to use this dataset to train a classification model, specifically a basic Convolutional Neural Network (CNN), to achieve good validation accuracy.

Path dataset

https://www.kaggle.com/datasets/ahemateja19bec1025/traffic-sign-dataset-classification

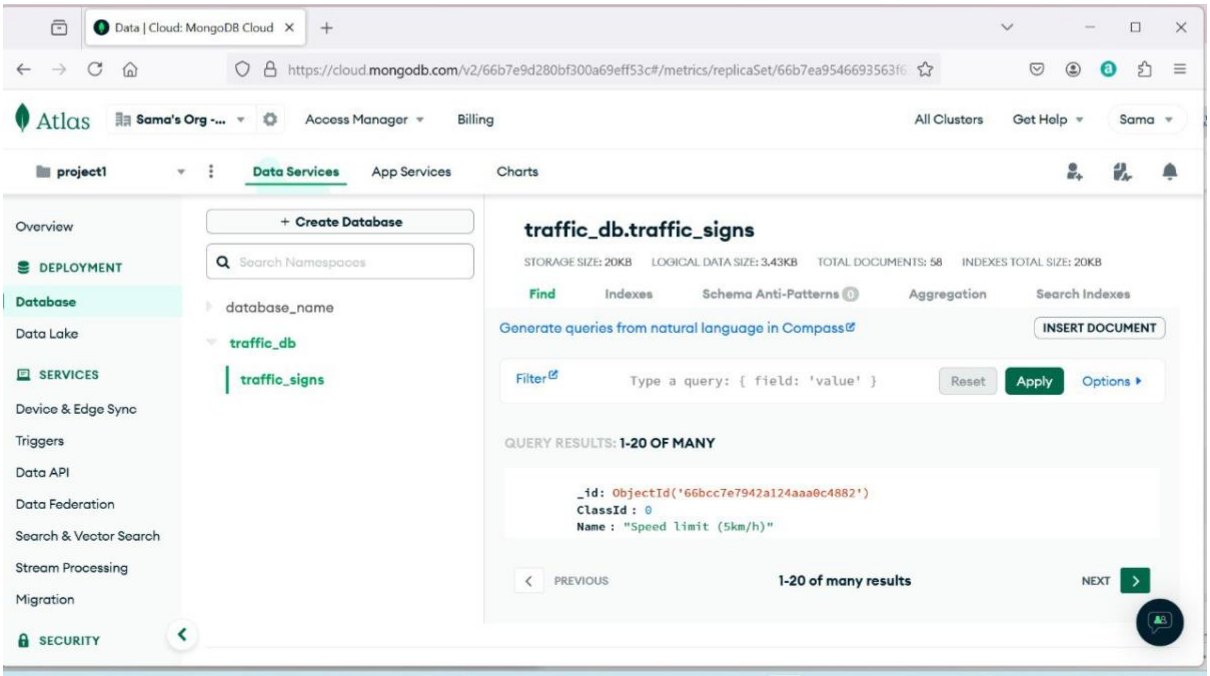
Methodology

1. **Data Collection**: Data is gathered and stored in MongoDB for efficient management.
2. **Exploratory Data Analysis (EDA)**: The dataset is analyzed to assess its characteristics and identify key patterns.
3. **Data Preprocessing**: The data is prepared for modeling through resizing, normalization, and augmentation.
4. **CNN Design and Implementation**: A Convolutional Neural Network (CNN) is designed and implemented to meet the project's specific requirements.
5. **Training and Validation**: The model undergoes iterative training and validation to optimize its performance.
6. **Testing**: The model is subjected to comprehensive testing on an independent test dataset to evaluate its accuracy and robustness.
7. **Evaluation**: The model's accuracy and generalizability are thoroughly evaluated to ensure its effectiveness.



MongoDB

We encountered several issues with MongoDB. Connecting to the database proved to be complex, and performance was significantly slow with large datasets. Additionally, there are concerns about security vulnerabilities, which could expose the system to potential attacks.



Result

Model Accuracy

- **Training Accuracy**: The training accuracy steadily increases over the epochs, reaching close to 1.0 by the end.
- **Validation Accuracy**: The validation accuracy also increases but shows more fluctuations compared to the training accuracy. It remains slightly lower than the training accuracy throughout.

Model Loss

- **Training Loss**: The training loss decreases significantly in the initial epochs and continues to decline, indicating that the model is learning well on the training data.
- **Validation Loss**: The validation loss decreases initially but starts to increase slightly after around epoch 5, suggesting potential overfitting as the model performs better on the training data than on the validation data.[Fig 1]

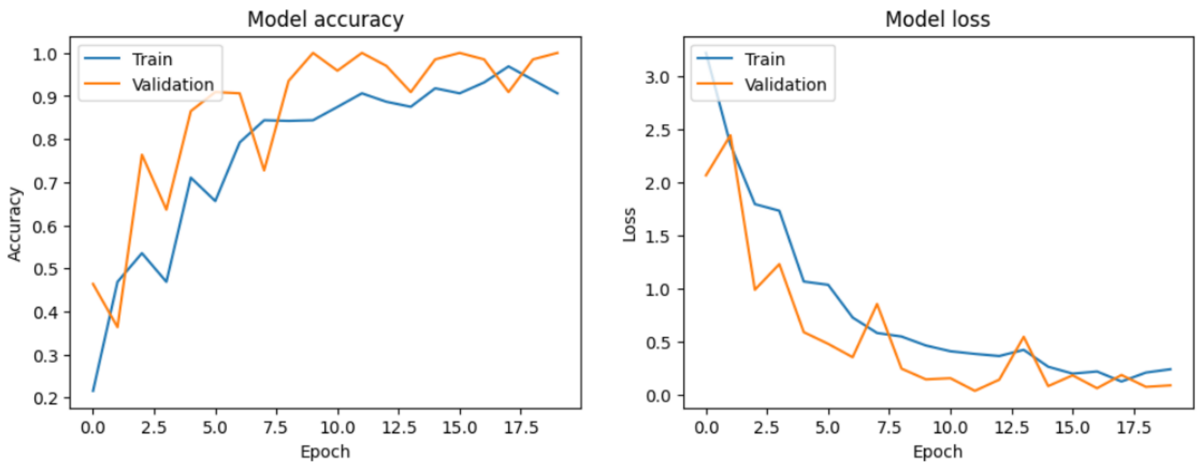


Fig1: large dataset

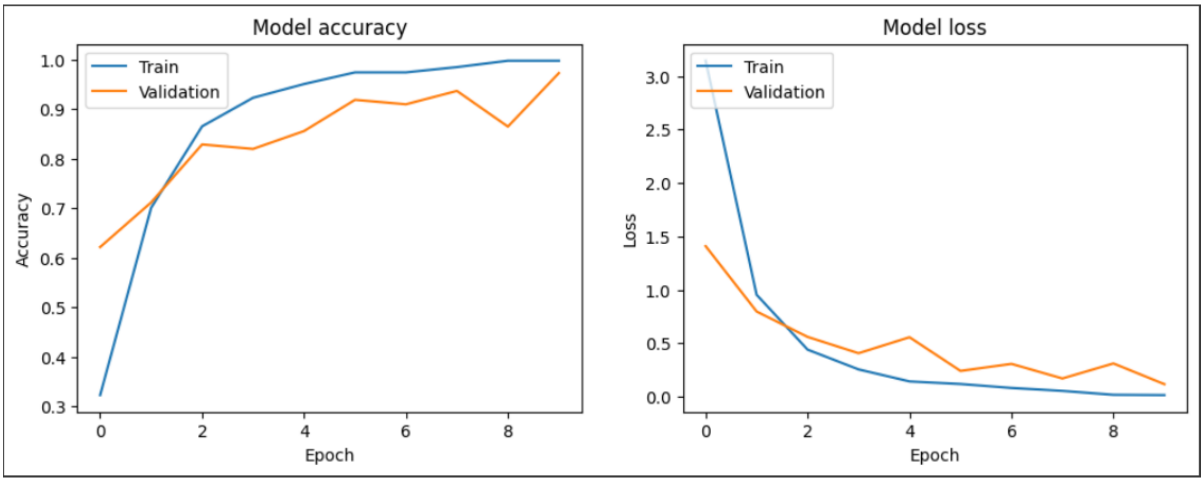


Fig 2: small dataset

Model Accuracy

- **Training Accuracy**: The training accuracy shows a steady increase over the epochs, reaching just over 0.9 by the end.
- **Validation Accuracy**: The validation accuracy also increases but with some fluctuations, remaining slightly lower than the training accuracy throughout.

Model Loss

- **Training Loss**: The training loss decreases significantly from just below 3.0 to just above 0.5, indicating that the model is learning well on the training data.
- **Validation Loss**: The validation loss decreases initially but shows some fluctuations, suggesting potential overfitting as the model performs better on the training data than on the validation data. [Fig 2]

What challenges did you face during data collection and how did you overcome

them?

• **Diversity of Traffic Signs:** One of the main challenges was ensuring that the dataset included a comprehensive variety of traffic signs from different regions and under different environmental conditions. This was crucial to avoid bias and ensure the model's ability to generalize across various scenarios.
• **Data Quality:** Variability in image quality, such as low resolution, motion blur, and varying lighting conditions, posed significant challenges. Inconsistent image quality could negatively impact the model's training and accuracy.

2. Why did you choose your specific model architecture?

The CNN architecture was chosen for traffic sign recognition due to its proven effectiveness in image tasks, ability to learn hierarchical features from raw data, robustness to variability in conditions, and flexibility in design. Its success in similar applications and compatibility with data augmentation further supports its suitability for accurate and efficient traffic sign detection.

3. How does the performance of your chosen model compare to alternative approaches, and what factors contributed to the differences?

The chosen CNN model generally outperforms alternative approaches like traditional machine learning models (e.g., SVMs) and simpler neural networks due to its ability to automatically learn and extract complex features from images. CNNs handle variations in lighting, angles, and occlusions better, leading to higher accuracy in traffic sign recognition.

4. What did you learn from storing the data in a database, and how did it impact your

workflow?

We encountered several issues with MongoDB. Connecting to the database proved to be complex, and performance was significantly slow with large datasets. Additionally, there are concerns about security vulnerabilities, which could expose the system to potential attacks.

5. If you were to extend this project, what additional features or improvements would

you consider?

Implement real-time traffic sign detection and recognition for applications in autonomous vehicles or driver assistance systems.

Enhance the model to detect and recognize multiple signs within a single image or video frame for better handling of complex traffic scenarios.