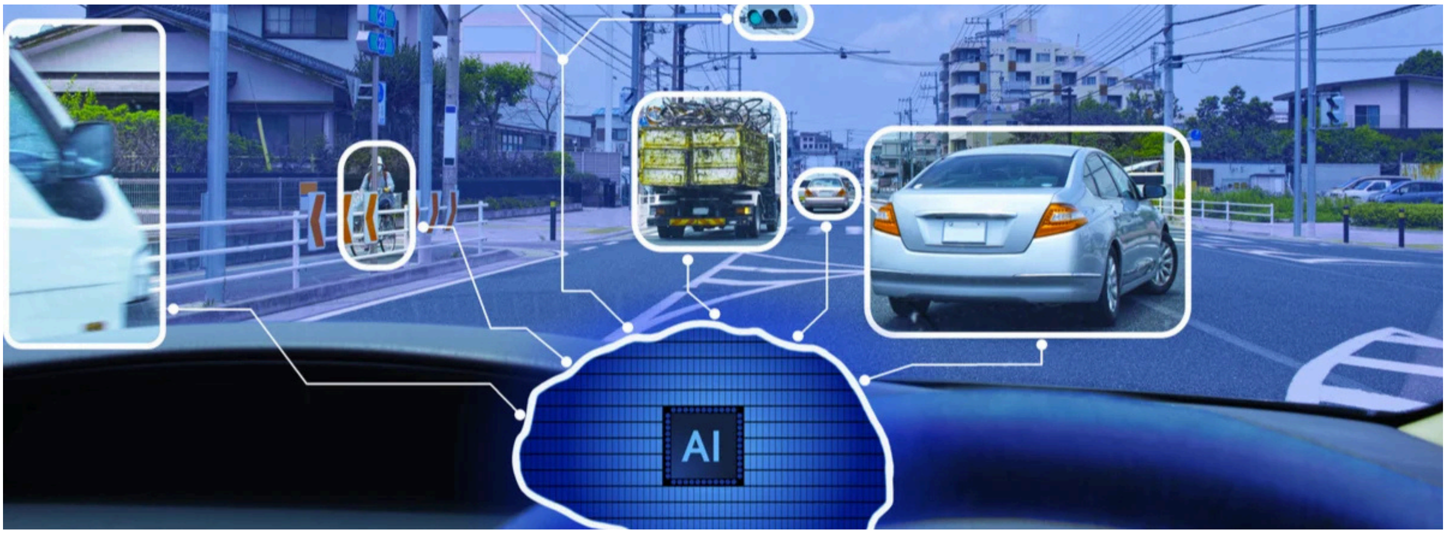
Autonomous Driving Vehicle Documentation (WEEK 1 & 2)



1. Overview and Objectives

The autonomous driving project aims to create a model that can interpret road environments and support safe, reliable driving. Key tasks include semantic segmentation, object detection, lane tracking, and traffic sign recognition. To develop these capabilities, a comprehensive preprocessing pipeline has been designed to prepare real-world driving images, ensuring they are optimized for training the deep learning model.

2. Dataset Characteristics and Challenges

2.1 Dataset Summary

- The dataset used in this project consists of real-world driving images collected in daylight conditions. These images, while reflective of normal driving environments, are currently without annotations for segmentation, object detection, lane markings, or traffic signs. The pipeline focuses on transforming these raw images into model-compatible formats, allowing them to be utilized in training.

2.2 Dataset Constraints

A notable constraint is that the dataset includes only daytime images, limiting its ability to generalize to more diverse scenarios :

- Absence of Nighttime and Low-Light Conditions : The dataset lacks nighttime images, which are essential for developing models capable of performing in low-visibility situations.
- Limited Weather Variability : Rainy or adverse weather conditions are absent, reducing the model's adaptability to unpredictable weather scenarios.

These limitations are addressed in the preprocessing steps, with additional plans to integrate varied lighting and weather conditions in future dataset expansions.

3. Image Preparation Workflow

3.1 Image Loading and Storage

Efficient data loading is the initial phase of preprocessing. This dataset includes only image files (no direct annotations), necessitating a streamlined approach to manage and load files.

Steps include:

- File Retrieval : Using `glob` to collect paths of all images within a specified directory.
- Memory Management : Large datasets require efficient memory handling techniques, such as batch processing and caching, to facilitate smooth data loading and processing.

3.2 Image Scaling and Standardization

To prepare images for the deep learning model, they are resized to a standard resolution of 1024x2048. This uniformity in resolution ensures consistent model input, which is essential for accurate training and performance evaluation.

A pre-trained DeepLabV3 model was applied for initial segmentation tasks, which allowed for a preliminary assessment of the dataset's suitability for semantic segmentation.

SEGMENTATION : **Google Colab**

3.3 Image Transformation Techniques

Since some images are initially grayscale, we employ color transformation techniques to make them resemble real-world visuals more accurately. The transformation process includes :

- **Converting Grayscale to Full Color :** Gray images are processed to match the color grading of typical real-time images.
- **Enhancing Lane Visibility :** Lane detection requires clear lane markings, often represented in white. Color adjustments ensure lane markings are distinguishable, improving the model's ability to recognize lane boundaries.

TRANSFORMATION : **Google Colab**

4. Evaluation and Outcomes

Initial Findings

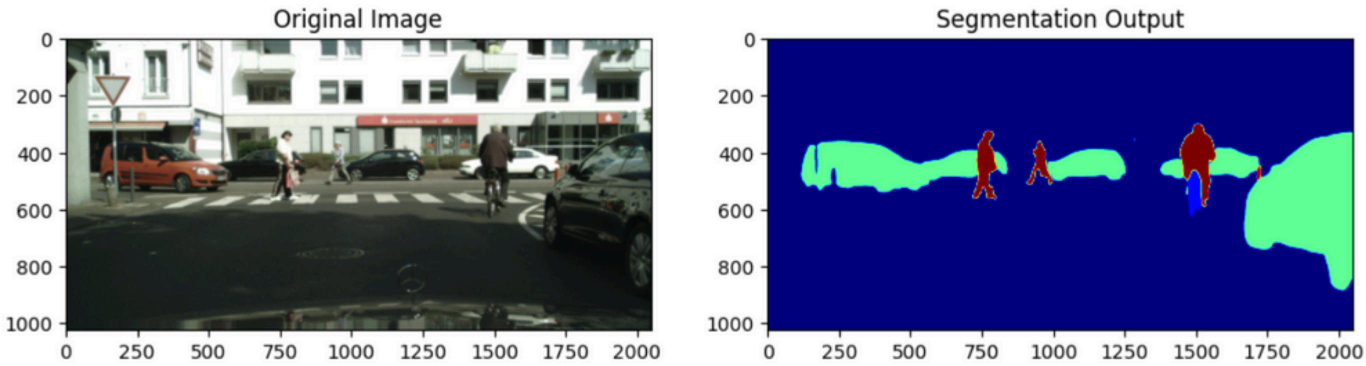
The pre-processing pipeline has enabled a clearer understanding of the dataset's strengths and limitations. Using a pre-trained segmentation model on the dataset provided insights into how well the model can distinguish road elements, aiding in object and lane detection.

Observed Challenges & Challenges encountered included:

- **Annotation Deficiency :** The lack of initial annotations required additional processing and transformation to make images usable.
- **Limited Scene Diversity :** Images primarily represent clear, daylight driving conditions, potentially limiting the model's adaptability to low-light or adverse weather conditions.
- **Transformation Benefits** : Through color enhancement, the dataset is now more suited to tasks like lane detection, where visual clarity is crucial. By standardizing color representation, lane boundaries are more readily detected, which strengthens the model's utility in autonomous driving systems.

5. RESULTS

The segmentation results highlight the model's effectiveness in distinguishing various elements within the scene. This is crucial for tasks such as object detection, lane detection, and understanding the driving environment. The model's performance on this sample indicates its potential applicability for real-time scenarios in autonomous driving systems.



The improved representation is especially beneficial for lane detection tasks, where lane markings are typically depicted in white. By ensuring that the colors are accurately represented, the model's ability to detect and segment lane markings effectively is enhanced, thereby improving the performance of autonomous driving systems.

6. FUTURE OBJECTIVES

5.1 Key Achievements

The preprocessing steps have allowed us to make substantial progress toward creating a dataset suitable for model training. With the color transformations, the dataset now supports essential tasks like lane detection and segmentation more effectively. However, future dataset diversification is required to address real-world conditions more comprehensively.

5.2 Upcoming Tasks

Task-Specific Training : Continue training the model on semantic segmentation, object detection, lane detection, and traffic sign recognition to enhance its functional scope.

Dataset Diversification : Expand the dataset to include nighttime images and varied weather conditions to increase the model’s real-world applicability.

Real-Time Optimization : As the model matures, focus on making it capable of real-time performance through model compression or optimized architecture.

User Testing and Feedback : Obtain feedback from users in real-world settings to assess model performance and identify areas for improvement.

7. REFERENCES

- Cityscapes Dataset

- How to convert a python numpy array to an RGB image with Opencv 2.4? - Stack Overflow
- Deeplabv3 | PyTorch

S.MANI GOWTHAM REDDY