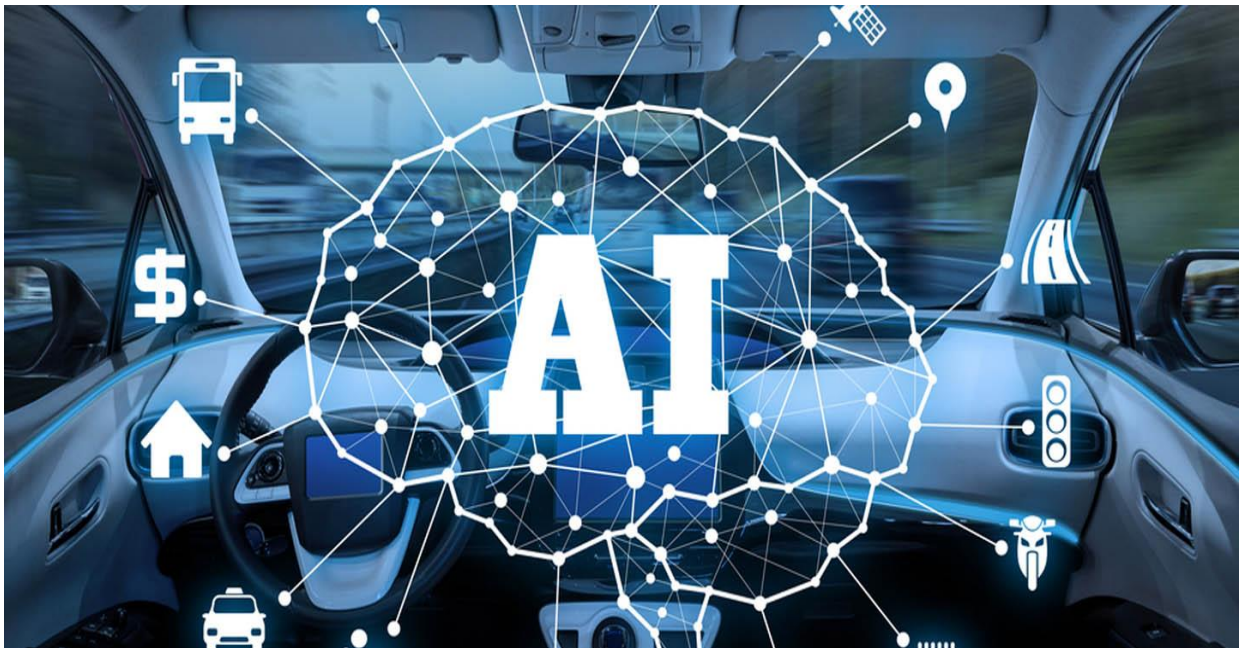
INFOSYS SPRINGBOARD INTERNSHIP 5.0

PROJECT REPORT



AI-Based Autonomous Driving Model Project

Submitted To:

Nitig Singh

(Spring Board Mentor)

Submitted By:

N.Deekshitha

➢ Introduction

The AI-based Autonomous Driving Model project aims to develop a framework capable of understanding urban environments using advanced machine learning techniques.

The project focuses on incorporating four key modules:

1. Semantic segmentation

- **Goal**: Classify each pixel of the image into categories like roads, vehicles, pedestrians, etc.
- **Popular Datasets** : Cityscapes, ADE20K, CamVid
- **Suggested Models** : UNet, DeepLabV3+, FCN (Fully Convolutional Networks)
- **Libraries** : TensorFlow, PyTorch

2.Object detection

- **Goal**: Detect vehicles, pedestrians, and other road objects in real time.
- **Approaches** : YOLO Fast detection, SSD (Single Shot Detector) Faster R-CNN: High accuracy
- **Dataset :** Cityscapes ,COCO, KITTI (automotive-related objects)
- **Tools :** TensorFlow Object Detection API, PyTorch with pre-trained models

3.Lane detection

- **Goal**: Identify and highlight lane markings on the road.
- **Approaches**: Classical CV approach , Deep Learning Models: SCNN (Spatial CNN)
- **Dataset** : Cityscapes , Tusimple Lane Detection, CULane
- **Libraries** : OpenCV , PyTorch/TensorFlow (for deep learning)

4.Traffic sign recognition

- **Goal:** Recognize and classify traffic signs to make decisions (e.g., speed limits, stop signs).
- **Dataset** : GTSRB (German Traffic Sign Recognition Benchmark)
- **Models :** CNN-based models: ResNet, MobileNet
- **Framework :** TensorFlow or PyTorch

Each module plays a vital role in enabling autonomous vehicles to perceive and respond effectively to their surroundings, ensuring safe and reliable operation

➢ Dataset Exploration :

The dataset has been successfully downloaded in its entirety and verified for completeness, ensuring that all required files are available for processing.

Directory Breakdown:

1. leftImg8bit

- **What it contains**:
    - This directory contains the **raw RGB images** of urban scenes in different cities.
    - Images are divided into **train**, **test**, and **val** folders, with subfolders corresponding to different cities (e.g., Aachen, Berlin).
    - Each image has a naming pattern like:

aachen_000000_000019_leftImg8bit.png

        - aachen: Name of the city where the image was captured.
        - 000000: Frame ID.
        - 000019: Sequence within the frame.
- **Usage**:
    - These images will be the **input** to your models for tasks such as **semantic segmentation** and **lane detection**.

---

2. gtFine (Ground Truth Annotations)

- **What it contains**:
    - The **gtFine** directory provides the **ground truth annotations** for the images in leftImg8bit.
    - For every input image, you'll find **4 corresponding ground truth files**:

        - ✓ aachen_000000_000019_gtFine_color.png
        - ✓ aachen_000000_000019_gtFine_labelIds.png
        - ✓ aachen_000000_000019_gtFine_instanceIds.png
        - ✓ aachen_000000_000019_gtFine_polygons.json

As part of the data exploration process, I thoroughly analyzed the Cityscapes dataset to understand its structure, completeness, and quality. By :

- ✓ **Verifying Dataset Structure and Contents**
- ✓ **Quality Check of Images**
- ✓ **Dimension Check**
- ✓ **RGB Intensity Distribution**
- ✓ **Sample Data Vizualisation**
- ✓ **Resizing and Preparation**

- **Notebook Provided for Easy Understanding**

I have documented the entire data exploration process in a **Jupyter Notebook**, which offers a step-by-step walkthrough of the dataset structure, verification, and preprocessing.

➢ Results:

- **<u>Week-1</u> :**

1. Verifying Dataset structure and contents:

```
Checking train folder...

Checking test folder...

Checking val folder...

Image Count Summary:
All files are available for train images: 2975 images (59.50%).
All files are available for test images: 1525 images (30.50%).
All files are available for val images: 500 images (10.00%).
```

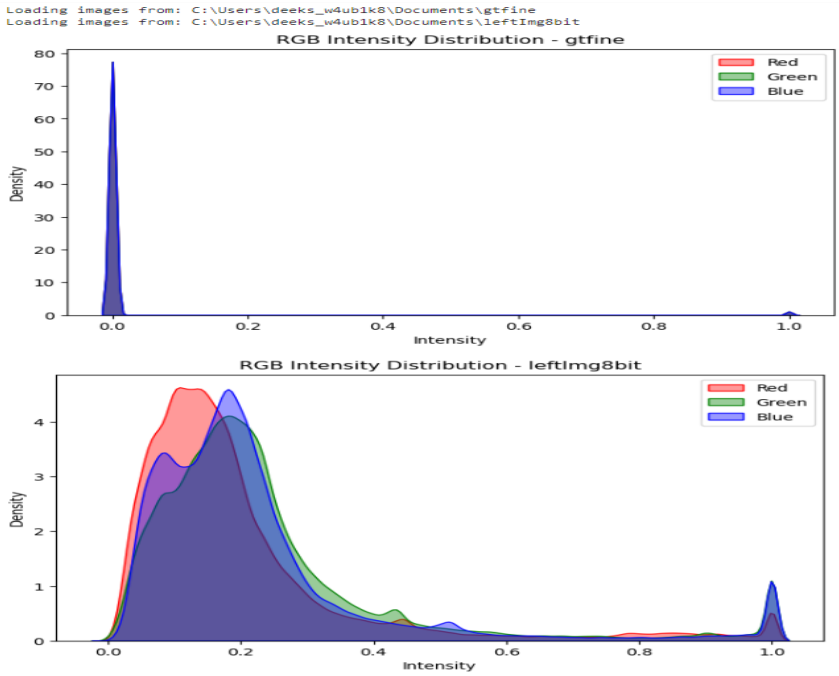
2.Quality Check of Images:

```
Total Images: 20000
Corrupted Images: 0 (0.00%)
Completely Black Images: 3068 (15.34%)
Completely White Images: 0 (0.00%)
All images are good!
```

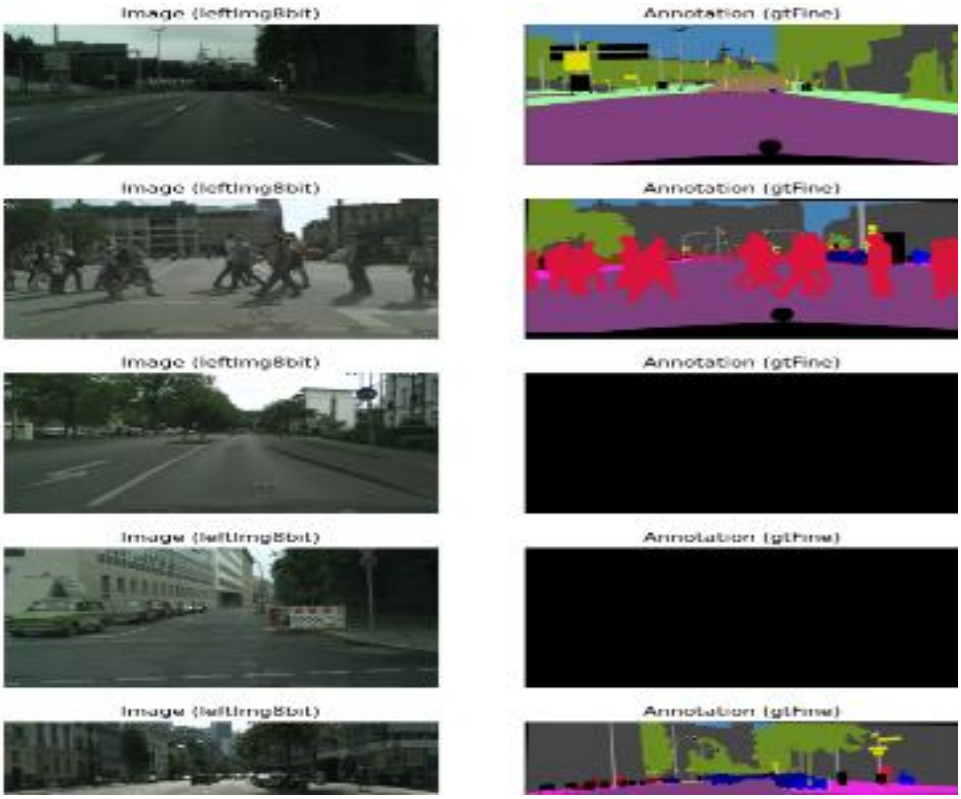
3.Dimension Check :

```
Total dimensions of 5000 images in 'C:\Users\deeks_w4ub1k8\Documents\leftImg8bit': 2048 x 1024 pixels.
Total dimensions of 15000 images in 'C:\Users\deeks_w4ub1k8\Documents\gtFine': 2048 x 1024 pixels.
```

4.RGB Intensity Distribution:



5.Sample Data Vizualisation:



Week-2:

1. Preprocessing: Resizing the images :

```
Resized and saved: C:\Gtfine\val\munster\munster_000173_000019_gtFine_color.png
Resized and saved: C:\Gtfine\val\munster\munster_000173_000019_gtFine_instanceIds.png
Resized and saved: C:\Gtfine\val\munster\munster_000173_000019_gtFine_labelIds.png
Copied JSON file: C:\Gtfine\val\munster\munster_000173_000019_gtFine_polygons.json
```

2. Dimension Check:

```
Total dimensions of 5000 images in 'C:\LeftImg': 512 x 512 pixels.
Total dimensions of 15000 images in 'C:\GtFine': 512 x 512 pixels.
```

➢ Conclusion :

In the initial phase of the project, we successfully explored and preprocessed the Cityscapes dataset, ensuring its readiness for model development.

Key activities included verifying the integrity of 20,000 images and their annotations, resizing the images to **512x512** pixels to meet model requirements, and splitting the dataset for effective training, validation, and testing.

Our analysis confirmed that the dataset is complete, with no corrupted or missing files, and well-suited for the planned modules: **semantic segmentation, object detection, lane detection, and traffic sign recognition**. We also provided a notebook for seamless dataset exploration and visualization to make the data more understandable.

With this solid foundation, we are now well-positioned to proceed with implementing the four modules in the coming weeks. The preparation ensures that all components are aligned to achieve optimal performance for the tasks at hand.

➢ Future Objectives (for the Next Two Weeks)

In the upcoming two weeks, the focus will shift towards implementing the foundational modules and ensuring smooth data integration. The specific objectives include:

1. **Developing Data Loading Pipelines:**
    - Implement efficient data loaders for images and annotations.
    - Ensure compatibility with deep learning frameworks (e.g., TensorFlow/PyTorch).

2. **Model Development for Each Module:**
    - **Semantic Segmentation:** Train models like U-Net or DeepLab for pixel-wise classification.
    - **Object Detection:** Utilize YOLO or Faster R-CNN to detect objects in urban environments.
    - **Lane Detection:** Implement segmentation-based techniques for detecting road lanes.
    - **Traffic Sign Recognition:** Use CNN-based models for classifying traffic signs accurately.

3. **Training, Validation, and Testing:**
    - Split the dataset into appropriate ratios and conduct initial model training.
    - Monitor metrics like accuracy, IoU, and loss for each task.
    - Adjust hyperparameters based on the model performance.

4. **Documentation and Progress Tracking:**
    - Keep refining documentation to track progress and challenges.
    - Provide visual results and observations to demonstrate intermediate outcomes.

These objectives will ensure smooth progression towards a well-structured AI-based autonomous driving model, aligning the work with project milestones and deliverables.

➢ References:

- ✓ https://www.cityscapes-dataset.com
- ✓ **ChatGPT**
- ✓ **Research article : Semantic segmentation of urbanenvironments Leveraging U-Net deeplearning model for cityscape image analysis**