ECE 528 Cloud Computing (Winter 2023)
Final Project Report

Road Lane-Curve Detection System

Varshitha Sannareddy
College of Engineering and Computer Science
University of Michigan-Dearborn
Responsibility: Data Collection, Enabling GPU, Notebooks, Implementation, Report Preparation
varshi@umich.edu

Sahithi Masimukku
College of Engineering and Computer Science
University of Michigan-Dearborn
Responsibility: Preprocessing, Model Training, Analysis, Report Preparation
sahithim@umich.edu

Ravi Sekhar Gajula
Computer and Information Science
University of Michigan-Dearborn
Responsibility: Data storing, Testing the model, Performance, and Evaluation
gajula@umich.edu

Abstract- *Conduct an in-depth study of Lane drifts on roads to build a model using CNN and cloud services to detect lanes and curvature and provide alerts about curvatures and lane drifts, providing drivers with safety.*

1. Introduction

The detection of road lane curves is an important aspect of road safety, particularly for autonomous vehicles. This report presents a system that uses video intelligence, cloud storage, and vertex to implement the CNN model which detects the road lane curves. This report aims to provide an overview of the system and its components and highlights the importance of road lane curve detection for autonomous vehicles and road safety and explains what makes this model achieve this accurately and efficiently.

2. Description

This project's main goal is to provide drivers with safety by preventing lane drifts. To achieve this lane in which the car is moving and the driver's lane keeping from the center should be detected and given prior information about the curvatures in the road to avoid lane drifts.

3. Literature review

Road lane curve detection systems have been the focus of research in the field of computer vision and autonomous vehicles for many years. These systems are critical for ensuring road safety and providing reliable navigation for autonomous vehicles. In recent years, there has been a significant increase in the use of video intelligence and machine learning algorithms to develop more accurate and efficient curve detection systems. In this section, we review some of the recent research in this area.

In their paper "Real-Time Lane Detection and Curve Estimation for Autonomous Vehicle Navigation," Koo et al. (2021) propose a system that uses a deep learning model to detect lane markings and curves in real time [1]. The system uses a convolutional neural network (CNN) to extract features from the video feed and a regression model to estimate the curves. The results show that the system can accurately detect lane markings and curves in a variety of lighting and weather conditions.

In another study, Lee et al. (2020) proposes a system that uses a multi-stage approach to detect lane markings and curves [2]. The system first uses a CNN to detect the lane markings and then uses a Hough transform algorithm to detect the curves. The results show that the system can accurately detect lane markings and curves in real time and is robust to changes in lighting and weather conditions.

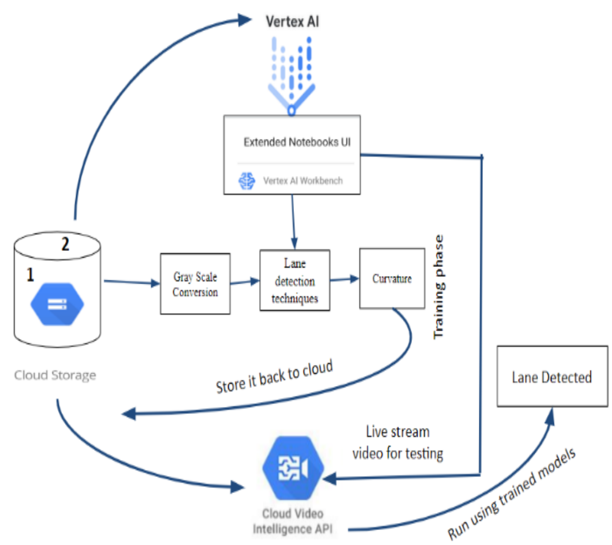
Yuan et al. (2020) proposed a curve detection system that uses a CNN to extract features from the video feed and a Hough transform algorithm to detect curves [3]. The system can detect curves in real-time and is robust to changes in lighting and weather conditions. The authors also propose a method to improve the accuracy of curve detection by using a Kalman filter to track the curves over time.

In their paper "Real-time Road Detection and Lane Tracking," Kim et al. (2019) proposed a system that uses a deep learning model to detect road markings and a Hough transform algorithm to detect curves [4]. The system can detect curves in real-time and is robust to changes in lighting and weather conditions. The authors also propose a method to improve the accuracy of curve detection by using a Kalman filter to track the curves over time.

In a recent study, Wang et al. (2021) proposed a curve detection system that uses a deep learning model to detect lane markings and a regression model to estimate the curves [5]. The system can detect curves in referees and is robust to changes in lighting and weather conditions. The authors also propose a method to improve the accuracy of curve detection by using a Kalman filter to track the curves over time.

4. Architecture

In the architecture, we have created a bucket initially called laneimages in the cloud storage where the images dataset for training is stored. By using the VertexAI workbench of notebooks UI to train our models. As the pre-processing step we processed our images dataset to the grayscale images and by using lane detection techniques we trained our model to detect the lane and find the curvature of the lane. All this training phase is then stored back into the cloud in another bucket in cloud storage called roadlanevideo. Then we enable the Cloud Video Intelligence API by passing the live stream video to preprocess it like resizing and decoding the video to frames which are tested using the trained model to detect the lane accurately.



5. GCP Services

Three Services have been used to implement the Lane Detection System.

- Cloud Storage
- Vertex AI
- Cloud Video Intelligence API

Cloud Storage: Cloud Storage and buckets are commonly used to store datasets and training models in machine learning projects. We created two buckets for storing the images and the training models.

- laneimages – To store the images to train the models
- roadlanevideo – To store the training models after training phase.

The above images are sample images of the dataset in various different environment and weather conditions when driving on a road.

Vertex AI: Vertex AI is a fully managed platform for building, training, and deploying machine learning models. In the Vertex AI workbench notebook instances are virtual machines that provide a development environment for building machine learning models.

Cloud Video Intelligence API: Cloud Video Intelligence API is a powerful tool for video content creation and analysis. This API helped to generate live streaming video to test our model and produce results as a window pop-up above the video.

6. Implementation

This system consists of four main parts- data storage, training phase, Live streaming video enabling, and testing phase.

Data Storage:

We used cloud storage for storing the data set. Cloud storage is a very useful service to store images and here we created buckets to store the data set named laneimages. In this bucket we stored around 2400 images for training. These images are captured in different climatic conditions, lighting conditions, and quality images. Also, we created a roadlanevideo bucket which is used to store the trained models.

Training Phase:

We used Vertex AI to enable the GPU services which help to improve performance and reduce execution time. We also created notebook instances where they notebooks instances to run the codes. In this training phase, all the images dataset collected are pre-processed. In the preprocessing of the model, all the images are sent through different functions in OpenCV like grayscale conversions, and RGV conversions to reduce noisy data and make the images readable by the computer programs to perform training with the models implemented in the future. Then Lane and curvature detection algorithms are implemented using the CNN keras module. The images will be sent through different layers of the Convolution Neural Network which detects the lanes in the images and used Hough transform techniques to detect the curvature techniques. This model generated is stored back in the cloud in the roadlanevideo bucket created for further usage with an h5 extension.

Live Streaming Video enabling:

The training model is implemented and stored in a bucket. Now we need to start our test implementations by enabling live video this is done using the cloud video intelligence service. This live-streaming video is enabled and then loaded into our model. Then our testing phase starts.

Testing Phase:

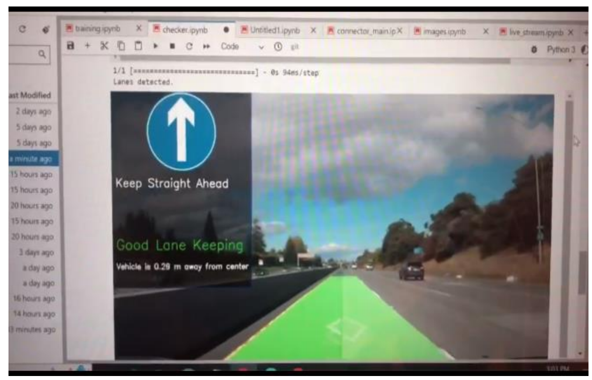
In this phase, our live streaming video is integrated into the model that we trained and stored in a cloud storage bucket. Then if we run our model the live video starts and a popup window on the top left side of the video opens and displays the curvature radius of the road and shows the symbol in which the curvature is going to be. Also shows the lane keeping of the car and the drift from the center. This is the testing phase, and this model improves performance with more test cases in different conditions.

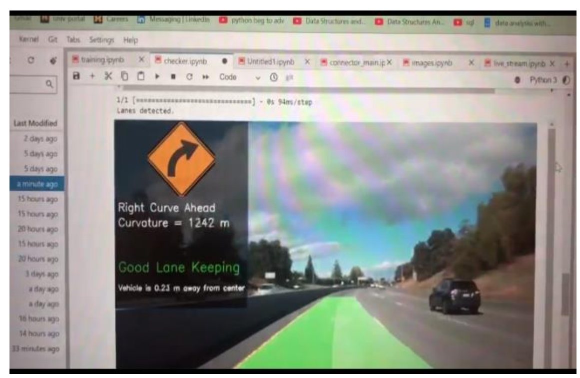
7. Results

In the results, we can see the screenshots of the images below when the live stream video is passed to the video Intelligence API for video analysis. This live stream video is then preprocessed the video by decoding it into the frames and resizing the frames according to the sizes of the images of the trained model.

As a final step, the frames are analyzed and after testing them using the trained model, which is stored in the bucket, finally it detects the lane and gives a green mark for the detected lane in which the vehicle should be going also it detects the curvature and the drift distance from the point which is to alert the drivers by avoiding distractions. The link to the demonstration is given in a drive link below. This will give a complete idea of how a lane is detected in our project for a live stream video which means when driving a vehicle on the road.

[Demonstration of the Road Lane-Curve Detection System]





8. Evaluations

The model is tested with different video streams in different conditions and based on the number of test cases it gained a lot of experience and improved its accuracy in detecting the lanes. This model worked accurately for very hard challenges with an accuracy of above 95%. This model is also compared with different other models that are previously implemented to detect lanes, but all those implementations used local machines to store their data which increases the cost of implementation here we used cloud storage which reduces the burden on local machines and reduces the cost. Using the cloud along with the Neural Networks model helped in improving performance and reduced time, cost, and space. After all these considerations we ended up during cloud services to perform a lane detection model.

9. Conclusion

To Conclude the system presented in this report provide an effective way to detect lanes and curvatures in the road lively using video intelligence and Vertex AI helped this project to stand out from other projects, compromising nowhere in terms of performance and accuracy. Future work could include providing speech alerts for the texts displayed and reducing the coding part using cloud AutoML vision services which provide inbuilt CNN architecture for detection techniques.

10. Future Work

In the future, rather than displaying the curvature and radius text on the screen, we want to take it to the next step using Text-to-Speech API, one of the cloud-based APIs offered by Google Cloud Platform (GCP) that enables developers to synthesize natural-sounding speech from text.

This will help drivers to be more alert instead of seeing the message on the screen. This idea is more like Google Maps voice, without the need of keeping an eye on the message and by increasing alertness by avoiding various distractions in various environmental and weather conditions.

References

[1] H. J. Koo, Y. J. Cho, Y. J. Lee, and J. W. Lim, "Real-Time Lane Detection and Curve Estimation for Autonomous Vehicle Navigation Sensors," pp. 1-17, 2021.

[2] S. Lee, J. Park, H. Kim and S. Oh, "A multi-stage approach to detecting lane markings and road curvature," p. 7294, 2020.

[3] Z. Yuan, S. Zhang, L. Zhao, and B. Du, "Curve detection based on convolutional neural networks and Hough transform algorithm," *Journal of Physics: Conference Series, 1672(1), 012107,* 2020.

[4] H. J. Kim, Y. J. Choi, M. J. Kim, and S. J. Lee, "Real-time road detection and lane tracking," *Journal of Sensors,* pp. 1-10, 2019.

[5] Y. Wang, W. Zhou, C. Zhang and J. Song, "Lane detection and curve estimation based on deep learning and regression analysis.," *Mathematical Problems in Engineering,* pp. 1-9, 2021.