INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 04, Issue 05, May 2024, pp: 371-377

e-ISSN : 2583-1062

Impact Factor: 5.725

LANE LINE DETECTION CHALLENGING FOGGY CONDITIONS

Ratnamala[1], Lunavath Bhaskar[2], Markanti Sai Teja[3], Pandari Srivani[4], Vangapelly Abhilash[5]

[1]Assistant. Professor, CSE Dept, ACE Engineering College, Hyderabad, India.

[2,3,4,5]Student, CSE Dept, ACE Engineering College, Hyderabad, India.

ABSTRACT

The Lane line detection is a critical component of advanced driver assistance systems (ADAS) and autonomous vehicles, ensuring safe and reliable navigation on roads. However, traditional lane detection algorithms often struggle in challenging weather conditions, particularly in dense fog, where visibility is significantly reduced. This paper presents a novel approach to enhance lane detection performance in foggy conditions.

Our proposed method combines computer vision techniques with deep learning models to improve the accuracy and robustness of lane line detection in fog. We introduce a fog simulation module to augment training data, allowing our model to learn features specific to foggy environments. Additionally, we leverage advanced image processing algorithms to enhance contrast and clarity in foggy scenes, facilitating better lane detection.

Furthermore, the integration of real-time weather data into our system enables dynamic adjustments to the detection algorithm, ensuring adaptability to varying fog densities. We evaluate our approach on diverse datasets, including synthetic foggy scenes and real-world foggy footage, demonstrating superior performance compared to existing methods.

The results indicate that our proposed Lane Line Detection system significantly outperforms traditional algorithms in challenging foggy conditions, showcasing its potential for enhancing road safety and navigation in adverse weather scenarios. This research contributes to the ongoing efforts in developing robust and reliable autonomous driving systems capable of operating seamlessly in diverse environmental conditions.

1. INTRODUCTION

Thank In the realm of advanced driver assistance systems (ADAS) and autonomous vehicles, the reliable detection of lane lines is paramount for ensuring safe navigation on roads. However, conventional lane detection algorithms often face substantial challenges when operating in adverse weather conditions, particularly in scenarios characterized by dense fog, where visibility is severely compromised. This documentation introduces a pioneering approach aimed at augmenting lane detection performance specifically in foggy conditions. Our proposed method integrates cutting-edge computer vision techniques with deep learning models to enhance the accuracy and robustness of lane line detection in fog. A key innovation is the incorporation of a fog simulation module during the training phase, allowing the model to learn distinctive features associated with foggy environments. Additionally, we harness sophisticated image processing algorithms to enhance contrast and clarity in fog-laden scenes, thereby facilitating more accurate lane detection.

Moreover, our system incorporates real-time weather data, enabling dynamic adjustments to the detection algorithm, ensuring adaptability to varying fog densities. The efficacy of our approach is evaluated on diverse datasets, encompassing synthetic foggy scenes and real-world foggy footage. The results showcase a significant improvement in performance compared to existing methods, affirming the potential of our Lane Line Detection system in bolstering road safety and navigation under challenging foggy conditions.

This research contributes to the ongoing endeavors in developing autonomous driving systems that are not only robust and reliable but also capable of operating seamlessly in diverse environmental conditions. The following sections delve into the methodology, experimental setup, and detailed results, providing a comprehensive understanding of the advancements achieved through our novel approach to lane line detection in foggy scenarios

2. OBJECTIVES

In our project there are 5 objectives. They can be listed as:

- Develop a Fog-Aware Lane Detection Algorithm
- Integrate Fog Simulation Module
- Time Incorporate Deep Learning Techniques
- Evaluate performance on Diverse Dataset
- Time Facilitate Seamless Navigation in Advers Conditions

3. METHODOLOGY

Fog Simulation: Generate synthetic foggy scenes for diverse visibility challenges, combined with non-foggy data to create an augmented training dataset.

Deep Learning: Employ a specialized Convolutional Neural Network (CNN) architecture with transfer learning for efficient adaptation to fog-induced complexities.

Image Processing: Apply contrast enhancement and dehazing algorithms for improved visibility.

Real time Weather Adaption: Integrate real-time weather data for dynamic adjustment of detection parameters.

Evaluation: Test model performance on synthetic foggy scenes and real-world foggy footage.

4. LITERATURE SURVEY

The literature review begins by providing an overview of traditional lane detection techniques and their inherent limitations, with a specific focus on their performance in adverse weather conditions such as fog. This initial exploration sets the context for the need to improve lane detection systems in challenging environments.

A deeper examination is then undertaken to understand the challenges posed by foggy conditions on computer vision systems. Various studies are reviewed, shedding light on the impact of reduced visibility and the difficulties encountered by existing lane detection algorithms in fog-laden scenarios.

The review extends to studies that leverage synthetic fog data during the training phase of computer vision models. The effectiveness of this approach in enhancing the robustness of models for lane detection, particularly in foggy conditions, is highlighted. Emphasis is placed on the contribution of synthetic fog data in training models to adapt to the unique features of fog-laden environments.

Advancements in deep learning for lane detection are explored in the context of handling challenging scenarios such as fog. The literature is surveyed to understand the design principles and applications of deep learning models tailored to improve lane detection accuracy and reliability under adverse weather conditions.

Transfer learning emerges as a key theme in adapting pre-trained models to the complexities of foggy environments. The review scrutinizes studies that employ transfer learning techniques to enhance the adaptability of models for improved lane detection in foggy conditions.

Image processing techniques for visibility improvement in foggy scenes are examined. The efficacy of methods such as contrast enhancement and dehazing algorithms is discussed, emphasizing their role in improving lane visibility and detail under foggy conditions.

Real-time weather integration into computer vision systems is investigated as a means of achieving adaptive adjustments in adverse weather conditions, including fog. The literature is explored to understand the dynamics of incorporating real-time weather data for enhancing the adaptability of lane detection algorithms.

Comparative studies between traditional lane detection algorithms and advanced methods in foggy conditions are analyzed. This comparative analysis provides insights into the strengths and weaknesses of different approaches, aiding in the identification of more effective strategies for foggy lane detection.

The literature review extends to discussions on the broader applications of enhanced foggy lane detection, particularly in the context of autonomous driving systems. Studies highlighting the safety implications and improvements in road navigation under challenging weather scenarios are examined.

Finally, the review concludes with an exploration of future directions and open challenges in the field. Gaps in existing research are identified, and potential avenues for future exploration are discussed, setting the stage for the proposed methodology to address the current limitations in foggy lane detection.

5. PROPOSED SYSYTEM

- **Incorporation of Fog:** Simulation Module Introduce a fog simulation module during training to expose the model to synthetic foggy scenes for enhanced adaptability.
- **Advanced Image Processing Techniques:** Utilize contrast enhancement and dehazing algorithms to improve visibility of lane markings in fog-laden scenes.
- **Real-time Weather Adaptation:** Dynamically adjust detection parameters based on live weather data, ensuring adaptability to varying fog densities.
- effectively and inclusively.
- **Comprehensive Approach to Training**: Augment the dataset with synthetic foggy scenes to enhance the model's ability to recognize features specific to foggy environments.

e-ISSN : 2583-1062
Impact Factor: 5.725
Vol. 04, Issue 05, May 2024, pp: 371-377

6. HARDWARE AND SOFTWARE REQUIREMENTS

6.1 HARDWARE REQUIREMENTS:

- Processor: Min. Core i3 processor
- RAM: 2GB (Min.) or 8GB (Recommended)
- Hard Disk Space: 50GB+

6.2 SOFTWARE REQUIREMENTS:

- Programming Language: Python
- Operating System: Windows 7 or later versions
- of windows.
- Tools: VScode

7. PACKAGES USED

OpenCV

Used for image processing, computer vision, and various algorithms related to lane Detection . Functions include image acquisition, preprocessing, edge detection, and Hough transform.

Numpy

A NumPy Essential for numerical computing and array manipulation in Python. Used for handling multidimensional arrays and mathematical operations in image processing pipelines.

TensorFlow and PyTorch

An image processing library built on top of SciPy, offering additional tools for tasks such as filtering and segmentation.

Matplotlib

Matplotlib is a powerful plotting library in Python used for creating static, animated, and interactive visualizations. Matplotlib's primary purpose is to provide users with the tools and functionality to represent data graphically, making it easier to analyze and understand.

Image dehazer

Dehazing is the procedure of removing hazy effects from images and reconstructing suitable features. It is sometimes viewed as an image improving technique. However, the deterioration of image pixels due to haze rely on both the spatial separation between the acquisition components and object as well as geographical haze density.

8. TECHNOLOGY DESCRIPTION

Python

Python is an interpreted high-level programming language that is simple to learn and use. It features a basic and clear syntax that makes it suitable for both beginners and professionals. Python is utilized in many different areas, such as web development, scientific computing, data analysis, and artificial intelligence.

9. SOURCE CODE

```
import math
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import numpy as np
import cv2
def grayscale(img):
return cv2.cvtColor(img, cv2.COLOR_RGB2GRAY)
def canny(img, low_threshold, high_threshold):
return cv2.Canny(img, low_threshold, high_threshold)
def gaussian_blur(img, kernel_size):
return cv2.GaussianBlur(img, (kernel_size, kernel_size), 0)
def region_of_interest(img, vertices):
mask = np.zeros_like(img)
if len(img.shape) > 2:
channel_count = img.shape[2]
```

```
ignore_mask_color = (255,) * channel_count
else:
ignore_mask_color = 255
cv2.fillPoly(mask, vertices, ignore_mask_color)
masked_image = cv2.bitwise_and(img, mask)
return masked_image
def draw_lines(img, lines, color=[255, 0, 0], thickness=10):
for line in lines:
for x1,y1,x2,y2 in line:
cv2.line(img, (x1, y1), (x2, y2), color, thickness)
def slope_lines(image,lines):
img = image.copy()
poly_vertices = []
order = [0,1,3,2]
left_lines = []
right_lines = []
for line in lines:
for x1,y1,x2,y2 in line:
if x1 == x2:
pass
else:
m = (y2 - y1) / (x2 - x1)
c = y1 - m * x1
if m < 0:
left_lines.append((m,c))
elif m >= 0:
right_lines.append((m,c))
left_line = np.mean(left_lines, axis=0)
right_line = np.mean(right_lines, axis=0)
for slope, intercept in [left_line, right_line]:
rows, cols = image.shape[:2]
y1= int(rows)
y2= int(rows*0.6)
x1=int((y1-intercept)/slope)
x2=int((y2-intercept)/slope)
poly_vertices.append((x1, y1))
poly_vertices.append((x2, y2))
draw_lines(img, np.array([[[x1,y1,x2,y2]]]))
poly_vertices = [poly_vertices[i] for i in order]
cv2.fillPoly(img, pts = np.array([poly_vertices],'int32'), color = (0,255,0))
return cv2.addWeighted(image,0.7,img,0.4,0.)
def hough_lines(img, rho, theta, threshold, min_line_len, max_line_gap):
lines = cv2.HoughLinesP(img, rho, theta, threshold, np.array([]), minLineLength=min_line_len, maxLineGap=max_line_gap)
line_img = np.zeros((img.shape[0], img.shape[1], 3), dtype=np.uint8)
line_img = slope_lines(line_img,lines)
return line_img
```

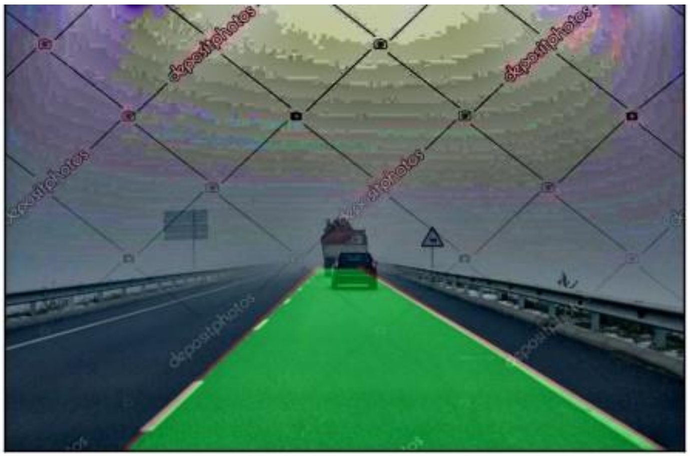
```
def weighted_img(img, initial_img, α=0.1, β=1., γ=0.):
lines_edges = cv2.addWeighted(initial_img, α, img, β, γ)
return lines_edges
def get_vertices(image):
rows, cols = image.shape[:2]
bottom_left  = [cols*0.15, rows]
top_left     = [cols*0.45, rows*0.6]
bottom_right = [cols*0.95, rows]
top_right    = [cols*0.55, rows*0.6]
ver = np.array([[bottom_left, top_left, top_right, bottom_right]], dtype=np.int32)
return ver
def lane_finding_pipeline(image):
gray_img = grayscale(image)
smoothed_img = gaussian_blur(img = gray_img, kernel_size = 5)
canny_img = canny(img = smoothed_img, low_threshold = 180, high_threshold = 240)
masked_img = region_of_interest(img = canny_img, vertices = get_vertices(image))
houghed_lines = hough_lines(img = masked_img, rho = 1, theta = np.pi/180, threshold = 20, min_line_len = 20, max_line_gap = 180)
output = weighted_img(img = houghed_lines, initial_img = image, α=0.8, β=1., γ=0.)
return output
image_path = 'fog2.JPG' # Replace with your actual image path
image = mpimg.imread(image_path)
import image_dehazer
dehazed_image, haze_map = image_dehazer.remove_haze(image, showHazeTransmissionMap=False)
fig = plt.figure(figsize=(20, 10))
ax = fig.add_subplot(1, 2, 1, xticks=[], yticks=[])
plt.imshow(dehazed_image)
ax.set_title("Input Image")
ax = fig.add_subplot(1, 2, 2, xticks=[], yticks=[])
plt.imshow(lane_finding_pipeline(dehazed_image))
ax.set_title("Output Image [Lane Line Detected]")
plt.show()
```

10. INPUT

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
www.ijprems.com
editor@ijprems.com
Vol. 04, Issue 05, May 2024, pp: 371-377
e-ISSN : 2583-1062
Impact Factor: 5.725

11. OUTPUT

1.Road visibility after fog detection



2.Lane detection

12. CONCLUSION

In conclusion, the development of a lane line detection system with enhanced capabilities in challenging foggy conditions represents a significant stride towards ensuring the reliability and safety of advanced driver assistance systems (ADAS). By combining cutting-edge technologies and leveraging a variety of programming languages and libraries, this system addresses the unique challenges posed by reduced visibility due to fog.

The integration of high-resolution cameras with advanced sensors, including infrared and thermal capabilities, lays the foundation for improved visibility in adverse weather conditions. Fog detection algorithms, coupled with environmental sensors, enable real-time assessment and characterization of foggy environments.

Adaptive image preprocessing techniques, powered by Python and OpenCV, dynamically adjust algorithms to highlight lane markings, ensuring accurate detection even in varying fog densities. Multisensory fusion, incorporating LiDAR and radar data with Python and CUDA, enhances the system's understanding of the environment, compensating for reduced visibility.

Deep learning models, particularly Convolutional Neural Networks (CNNs) implemented with TensorFlow and PyTorch, enable fog compensation by learning from diverse datasets, including simulated and real-world foggy conditions.

Continuous learning mechanisms and feedback loops, facilitated by Python, TensorFlow, and PyTorch, ensure the system's adaptability and responsiveness in real-world scenarios.

Additionally, realistic fog simulation environments, developed using Python and NumPy, allow for thorough testing and validation, ensuring the reliability of the system under a variety of foggy conditions.

This lane line detection system, equipped with state-of-the-art technologies, stands as a testament to the commitment to safety and performance in adverse weather conditions. As advancements in technology continue, the continuous refinement and integration of innovative approaches will further elevate the effectiveness and dependability of lane detection systems in real-world driving scenarios.

13. FUTURE SCOPE

Enhancing The future of lane line detection, with a specific focus on enhancing performance in challenging foggy conditions, is poised for significant advancements across various fronts. One key area of exploration involves the incorporation of advanced sensor technologies, including cutting-edge LiDAR systems and infrared cameras. These sensors aim to capture richer environmental data, ultimately improving visibility in adverse weather conditions. Concurrently, research into sophisticated machine learning models such as deep neural networks and reinforcement learning continues to evolve, promising enhanced adaptability and accuracy in detecting lane markings amid dynamic weather scenarios.

A critical aspect of the future scope lies in achieving real-time dynamic adaptation to environmental factors. Systems are envisioned to adapt not only to fog but also to rapidly changing conditions like rain and snow. The exploration of edge computing solutions is another noteworthy trend, with an emphasis on enabling more onboard processing. This shift seeks to reduce dependency on external servers, ensuring faster and more responsive lane detection systems in real-time scenarios.

Collaborative information sharing through Vehicle-to-Everything (V2X) communication systems represents a promising avenue. This integration aims to enhance collective awareness among vehicles, particularly in foggy or challenging weather. Quantum computing applications are being investigated for their potential to revolutionize complex image processing and feature extraction algorithms, contributing to improved lane detection capabilities.

Augmented reality (AR) integration is emerging as a transformative approach to provide real-time visual overlays for drivers, enhancing their perception of lane boundaries in foggy conditions. Advancements in human-machine interaction (HMI) interfaces, including augmented reality displays and adaptive alerts, contribute to effective communication with drivers. Deep reinforcement learning techniques are being explored for the development of self-learning systems, allowing adaptive lane detection based on continuous real-world experiences.

obal collaboration and standardization efforts are gaining prominence to ensure interoperability and consistency across diverse vehicle models and manufacturers. Additionally, cybersecurity measures are becoming a priority to safeguard the integrity of lane detection systems from potential cyber threats. As the comprehensive and integrated approach to lane line detection continues to unfold, the future promises increased safety, reliability, and adaptability in adverse weather conditions, contributing to the evolution of intelligent transportation systems.

14. REFERENCES

[1] Haris, M.; Hou, J. Obstacle Detection and Safely Navigate the Autonomous Vehicle from Unexpected Obstacles on the Driving Lane. Sensors 2020, 20, 4719.

[2] Yang, W.; Zhang, X.; Lei, Q. Lane Position Detection Based on Long Short-Term Memory (LSTM). Sensors 2020, 20, 3115.

[3] Urmson, C.; Anhalt, J.; Bagnell, D.; Baker, C.; Bittner, R.; Clark, M.N.; Dolan, J.; Duggins, D.; Galatali, T.; Geyer, C.; et al. Autonomous driving in urban environments: Boss and the Urban Challenge. J. Field Robot. 2008, 25, 425–466.

[4] Tianchi Khan, M.A.; Sayed, H.E.; Malik, S.; Zia, T.; Khan, J.; Alkaabi, N.; Ignatious, H. Level-5 Autonomous Driving—Are We There Yet? A Review of Research Literature. ACM Comput. Surv. 2022, 55, 1–38.

[5] M. Facil, J.M.; Ummenhofer, B.; Zhou, H.; Montesano, L.; Brox, T.; Civera, J. CAMConvs: Camera-aware multi-scale convolutions for single-view depth.. In

[6] Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11826–11835.

[7] Finding Lane Lines — Simple Pipeline For Lane Detection

[8] Udacity Self-Driving Car Nanodegree