This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Lane Detection in Autonomous Vehicles: A Systematic Review

N. J. Zakaria[1], M. I. Shapiai[1], R. A. Ghani[1], M. N. M. Yasin[2,3], Member, IEEE, M. Z. Ibrahim[4], and N. Wahid[5]

[1]Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia
[2]Advanced Communication Engineering (ACE), Centre of Excellence, Universiti Malaysia Perlis (UniMAP), Kangar, Perlis 01000, Malaysia
[3]Faculty of Electronic Engineering Technology, Universiti Malaysia Perlis (UniMAP), Arau, Perlis 02600, Malaysia
[4]Faculty of Electrical and Electronic Technology Engineering, Universiti Malaysia Pahang, Pekan, Pahang
[5]College of Engineering, Universiti Teknologi Mara, Dungun, Terengganu, Malaysia

Corresponding author: Mohd Ibrahim Shapiai (e-mail: md_ibrahim83@utm.my).

This research was funded by UNIVERSITI MALAYSIA PERLIS and Ministry of Higher Education, grant number FRGS/1/2020/ICT09/UNIMAP/02/2 and R.K130000.7343.4B512 – CRG 30.4: Survivors Identification and localization for SAR robotics system using machine learning and supported by the University Fundamental Research Grant scheme RDU200318 which are awarded by Universiti Malaysia Pahang (UMP) Malaysia.

ABSTRACT Over the years, the number of vehicular traffic on the road has dramatically increased, causing the rate of road accidents also to increase. As a result, research on vehicle development and safety has significantly grown. One of the essential systems in autonomous vehicles for ensuring a secure circumstance for drivers and passengers is the Advanced Driver Assistance System (ADAS). Adaptive Cruise Control, Automatic Braking/Steer Away, Lane-Keeping System, Blind Spot Assist, Lane Departure Warning System, and Lane Detection are examples of ADAS. Lane detection displays information specific to the geometrical features of lane line structures to the vehicle's intelligent system to show the position of lane markings. This article reviews the methods employed for lane detection in an autonomous vehicle. A systematic literature review (SLR) has been carried out to analyze the most delicate approach to detecting the road lane for the benefit of the automation industry. One hundred and two publications from well-known databases were chosen for this review. The trend was discovered after thoroughly examining the selected articles on the method implemented for detecting the road lane from 2018 until 2021. The selected literature used various methods, with the input dataset being one of two types: self-collected or acquired from an online public dataset. In the meantime, the methodologies include geometric modeling and traditional methods, while AI includes deep learning and machine learning. The use of deep learning has been increasingly researched throughout the last four years. Some studies used stand-alone deep learning implementations for lane detection problems. Meanwhile, some research focuses on merging deep learning with other machine learning techniques and classical methodologies. Recent advancements imply that attention-mechanism has become a popular combined strategy with deep learning methods. The use of deep algorithms in conjunction with other techniques showed promising outcomes. This research aims to provide a complete overview of the literature on lane detection methods, highlighting which approaches are currently being researched and the performance of existing state-of-the-art techniques. This review yields a valuable foundation on lane detection techniques, challenges, and opportunities and supports new research works in this automation field. For further study, it is suggested to put more effort into accuracy improvement, increased speed performance, and more challenging works on various extreme conditions in detecting the road lane.

INDEX TERMS Lane detection, autonomous vehicle, systematic literature review, geometric modelling, deep learning, machine learning.

I. INTRODUCTION

According to a World Health Organization (WHO) report published in June 2022, approximately 1.3 million people die yearly from road traffic accidents [1]. As a human driver, it is hard to remain in the correct lane and to keep the following proper gap with the front vehicle, as the driver

IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

needs to focus on the road for an extended time. Moreover, humans are prone to driver fatigue, sleepiness, inattention, and drowsiness. Besides that, using technologies in vehicles such as smartphones, entertainment, and navigation systems may interrupt the driver and compromise safety while driving. Therefore, the costs of road traffic accidents to society are expensive in terms of human injury and economic loss. The development of passive and active safety systems for automobiles has resulted from the abovementioned concern. Seat belts and airbags are examples of passive safety systems [1].

These were developed to decrease the risk of injury to the driver and passenger from the impact of accidents. These systems have become the standard safety gear for vehicles but are only utilized after accidents occur, but it would be far better if the casualties were entirely prevented. As a result, active safety technologies are becoming a talking point among automakers and researchers [2]. The evolution of autonomous cars started in Europe around 1986. At this time, several car manufactures and research institutes initiated a series of innovative vehicle safety projects and research to obtain practical solutions for urban traffic problems. For instance, the European Union introduced the Generic Intelligent Driver Support (GIDS) project under the Dedicated Road Infrastructure for Vehicle Safety in Europe (DRIVE) [3]. This massive Intelligent Vehicle project aims to assist the driver's identification and estimation of traffic danger and, in turn, assign a system to deal with specific hazards. The essential goal of the development system in an autonomous vehicle is to assist drivers in identifying driving risks and ensuring extra safety and comfort for the driver and passengers in the car.

The Advanced Driver Assistance System (ADAS) is one of the essential systems in autonomous vehicles for making the driving environment safer for drivers and passengers. ADAS aims to reduce driver error by helping to avoid vehicle collisions, increase traffic efficiency, and enhance transportation development. Adaptive Cruise Control [4], Automatic Braking/Steer Away [5], Lane-Keeping System [6], Blind Spot Assist [4], Lane Departure Warning System [7], and Lane Detection [8] are several examples of the ADAS module.

The lane is a traffic sign that divides a road in the traffic system and guarantees that automobiles are driven safely and effectively. Lane detection is a technique for automatically detecting road markers to ensure that cars stay in their assigned lane and do not collide with the vehicle in other lanes. It has played a part in autonomous driving. As a result, accurate lane detection allows the autonomous vehicle to make multiple decisions and judgments about its location and state and to ensure safe driving [9]. Lane detection algorithms are difficult to use because of the wide variety of lane markers, the complex and changing road conditions and environment, and the lane's inherent slenderness [10]. Hence, the significant research has developed reliable lane detection algorithms [11].

To solve this problem, various hand-crafted methods, including geometric modeling and traditional approaches, have been used to detect the lane markers. Most conventional detection strategies adhere to pipelines, which typically consist of image pre-processing, feature extraction, lane model fitting, and line tracking. The purpose of image pre-processing is to reduce the quantity of noise in the image. Next, the features of lanes are utilized in the feature extraction process, extracting areas that are lanes. After that, the lane model is fitted and tracked via various selected methods. Several previously applied techniques for feature extraction, such as Inverse Perspective Mapping (IPM)/Perspective Transform, filtering technique, edge detection-based technique, image district extraction, morphological operators, neighborhood searching-based feature points, grayscale, thresholding, clustering, heterogeneous operators, and sliding window. These techniques help reduce noise and make it easier to extract lanes. Next, the lane model is typically fitted with a line segment detector (LSD) and fitting-based methods like B-spline, quadratic, polynomial, hyperbolic, and the least square methodology. After that, the Kalman filter, lane classification, and the parabola equation are the three methods utilized most frequently in tracking road lane detection. In addition, tracking is used as the post-processing step to compensate for fluctuations in the illumination [11]. Therefore, tracking also helps with incorrect occlusion detection induced by inadequate lane markers [12]. However, the traditional methods involve a process that is not only more difficult but also hand-crafted, which results in a significantly longer processing time.

The recognition of lane markings has grown significantly more accessible, faster, and more effective in recent years, given the proliferation of Artificial Intelligence (AI) technologies. In addition, there is no longer a need to employ hand-crafted procedures. AI is the simulation of human intelligence processes by computers, most notably computer systems. Machine Learning (ML) and Deep Learning (DL) are the two primary categories that may be used to classify most of the AI approaches used in lane detection. The DL approach has become more popular than ML due to its effective performance in either classification or detection, utilizing image frames as input to the network algorithm. This is the primary reason for the rise in popularity of the DL method. Bayesian Classifier, Haar Cascades, Extreme Learning Machine (ELM), Support Vector Machine (SVM), and Artificial Neural Network (ANN) are some examples of the ML algorithms that are utilized in this field.

Meanwhile, the use of the DL technique as a stand-alone approach was suggested by some researchers, while many others advised integrating this method with another approach. The goal of the integration of this network is to improve the effectiveness of the network in challenging conditions when it comes to identifying the lane mark. Other than that, DL is combined with geometric modeling. DL merged with ML and DL combined with DL are all examples of the integration of another method. Aside from that, in recent times, a new integration idea for this method has been offered, and it involves merging DL with an attention mechanism. This is the latest state-of-the-art technique that has been proposed, and there is room for further investigation.

A new study addresses this need by thoroughly examining the implementation of various techniques in lane detection. Thus, this paper lays a solid foundation for lane detection methodologies, challenges, and opportunities and lays the groundwork for more research on this subject of automation. Furthermore, this study provides an overview of what has been done in the last four years of literature published related to the method used to detect the road lane. In addition, the study focused on answering specific issues about the collecting data equipment, lane detection learning algorithms/network topologies, and the dataset used for lane detection systems. This research shows the difficulties in implementing learning algorithms and determining future research areas. It also serves as a resource for researchers and professionals working in the lane detection sector, assisting them in the latest approaches or developing new lane detection frameworks for accuracy enhancement and performance under various scenarios.

The rest of the article is arranged as follows: The research questions, review protocol consisting of search sources, search terms, inclusion and exclusion criteria, and literature collection are all described in Section II. The literature that was chosen and analyzed statistically is presented in Section III. Section IV summarises the literature to address each question, constructively evaluates the outcomes, and highlights key points. Finally, Section V concludes the study with some suggestions for further research.

II. SYSTEMATIC LITERATURE REVIEW

The writing for this paper consists of planned, conducted, and observed processes, as shown in Figure 1. First, the planning phase has clarified the research questions and review protocol containing the publications sources, keywords search, and selection criteria. The next stage is conducting a phase related to analyzing, extracting, and synthesizing the literature collection. The last step, the observed stage, contains the review results that address the research questions and the objectives described.

A. RESEARCH QUESTIONS

This review's main objective is to determine the trend of the method implemented for lane detection in the autonomous vehicles field and the achievement of the current latest techniques. Other than that, to look into the valuable foundation on the methods, challenges, and opportunities. Thus, providing the state-of-the-art knowledge to support new research works in this computer vision and automation field. Hence, the three research questions (RQs) described have been stated as follows:

1) What techniques have been implemented for lane detection in an autonomous vehicle?
2) What equipment is being used to collect the dataset?
3) What dataset was applied for the network training, validation, and testing?

The focused approach has been adopted while scanning the literature. First, each article was reviewed to see if it answered the earlier questions. The information acquired was then presented comprehensively to accomplish the vision of this article.

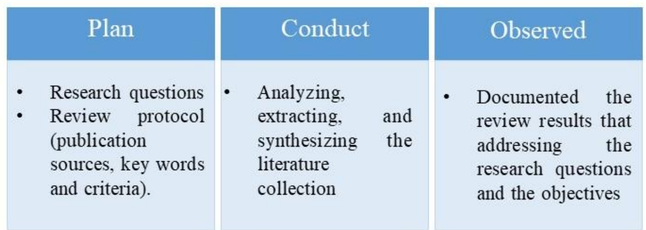

FIGURE 1. **Process of Systematic Literature Review. It consists of three stage which are plan, conduct and observe.**

B. REVIEW PROTOCOL

The following are the literature search sources, search terms, and inclusion and exclusion selection criteria. Also, the technique of literature collection used for this SLR:

1) SEARCH SOURCES

Scopus, IEEE Xplore, Web of Science, and Springer Link were chosen as the databases from which the data was extracted.

2) SEARCH TERMS

'Lane detection' and 'autonomous vehicle' are two prominent search terms used to investigate the topic. The terms 'lane detection' can be searched using different words. The 'OR' operator was used to choose and combine the most relevant and regularly used applicable phrases. For example, the search phrases 'lane detection,' 'lane tracking,' and 'lane segmentation' were discovered. The 'AND' operator combined individual search strings into a search query. Figure 2 shows the complete search query for each of the databases. The databases include Scopus, Web of Science, IEEE Xplore, and Springer Link.

3) INCLUSION

The study covered all primary publications published in English that used the approach for lane detection, tracking, segmentation, or any other task related to detecting the road lane. There were no constraints on subject categories or time frames for a broad search spectrum. The selected articles are published for four years, from 2018 to 2021. In addition, Journal papers, conference proceedings, and book sections on the subject were included in the research.

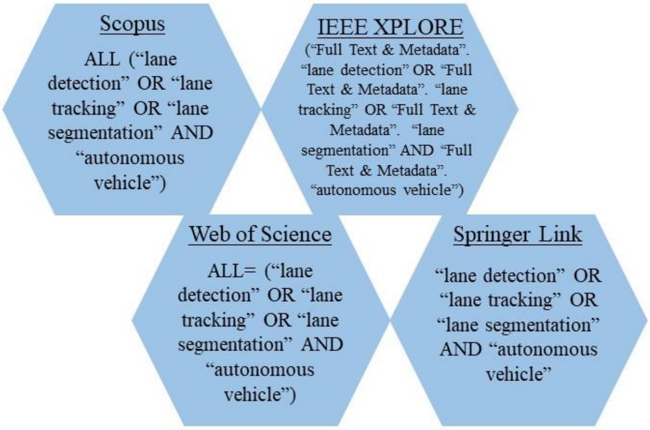

FIGURE 2. Search queries for each of the databases. The databases include Scopus, Web of Science, IEEE Xplore and Springer Link.

4) EXCLUSION

Articles written in languages other than English were not considered. Therefore, the exclusion criteria included short papers, such as abstracts or expanded abstracts, and survey/review papers.

C. LITERATURE COLLECTION

The literature search was carried out by providing the search strings for each database, as shown in Figure 3. These search keywords resulted in a total of 435 publications being returned. Next, each database's search results were evaluated using predetermined inclusion/exclusion criteria. The initial screening excluded review articles and non-English journals. After that, each manuscript was assessed based on its title, abstract, and a short read of the content to determine if it should be accepted or rejected. The number of articles was reduced to 158 after this filtration. Next, after removing duplicate papers, 114 publications were included in the full-text review. For reasons such as publications that are not available as full text and similar to the previous articles by the same author, just a small number of enhancements are also excluded. Then, 102 studies were chosen to be included in this SLR. As discussed above, the steps to obtain the publications related to this SLR have been presented in the form of PRISMA. The Preferred Reporting Items for Systematic Review and Meta-analysis are shown in Figure 3. (PRISMA) [13].

III. RESULTS

Table I lists the chosen publications, the year of publication, source title, and the number of citations. About 102 publications have been listed in Table I with the state of references. The lists included journals, conferences, and book chapters. Figure 4 depicts the publishing distribution from 2018 to 2021. Every year, a growing tendency in the literature is visible in the yearly distribution displayed in Figure 4. For example, in 2018, about 16 papers were published, and 25 articles were published in 2019. Meanwhile, 29 and 32 papers were published in 2020 and 2021. Next, from 2018 to 2021, 48 articles were published in conference proceedings, 44 in journals, and ten as book chapters, as shown in Figure 5. For example, in 2018, 11 conferences, three journals, and two book chapters were published.

Meanwhile, for the coming year, 2019, 16 conferences, eight journals, and only one book chapter on-road lane detection have been published. Next, 14 conference papers, 12 journals, and three book chapters have been published for 2020. Finally, the number of conferences published in 2021 is down from the previous year, when just seven articles were released. In the meantime, journal publications have climbed to 21, with four book chapters scheduled for release in 2021. Table II shows the distribution of papers in journals. Sensors journal ranks first with five publications, followed by Journal of Ambient Intelligence and Humanized Computing, International Journal of Advanced Robotic Systems, Journal of Electrical Engineering and Technology, Multimedia Tools and Applications, and IEEE Access, which ranks second with two publications per article.

Table III indicates the publications of lane detection in conferences. The tables show that the Advances in Intelligent Systems and Computing conference ranks first with five publications, followed by ACM International Conference Proceeding Series, 2nd International Conference for Emerging Technology, INCET 2021, Chinese Control Conference, CCC, IET Conference Publications, and 2018 6th International Conference on Control Engineering and Information Technology, CEIT 2018 which ranks second with two publications per conference.

Table IV shows the publications of lane detection in book chapters. There are ten book chapters which are Advanced Structured Materials, Lecture Notes on Data Engineering and Communications Technologies, Transactions on Computer Systems and Networks, Image and Graphics, Lectures Notes in Network and Systems, Computational Intelligence in Data Science, Databases and Information Systems, Lecture Notes in Computational Vision and Biomechanics, Image and Video Technology and Computational Science and Technology.

IV. DISCUSSION

To answer the RQs, each publication was thoroughly examined with the necessary data extracted. Consists of the primary approach, the type of dataset used in the study, whether self-collected or acquired from an online dataset. Each publication focuses on the dataset's collection and preparation for network training and testing. The findings for each RQ in their respective sections are as follows:

A. WHAT METHODS HAVE BEEN APPLIED FOR LANE DETECTION IN AUTONOMOUS VEHICLES?

This section explores several related studies on detecting road lane markers. The strategies for lane detection can be categorized into two methods based on past research: Geometric modeling/traditional approaches for lane detection and ii) Artificial Intelligence-based techniques. These are outlined in further detail below:

1) GEOMETRIC MODELLING/TRADITIONAL METHODS

The pipelines used by most traditional detection algorithms comprise image preprocessing, feature extraction, lane model fitting, and line tracking. Image preprocessing aims to remove some of the noise from the image. Feature extraction employs lanes' features to extract lane-like areas. The lane model is then fitted and tracked using a variety of methods. Feature detection is an essential lane detection algorithm that affects performance [10]. As a result, the preprocessing image phase is required in many traditional methodologies for determining the quality of features for lane detection tasks. The construction of an area of interest (ROI), image augmentation for extracting lane information, and removing non-lane details are all part of image processing. The ROI extraction method efficiently reduces redundant information in the image preprocessing section by selecting the lower portion of the image [11]. Several studies have created ROIs using vanishing point detection techniques [11], [14]. Furthermore, ROI creation minimizes image noise, although it is not resistant to shadows or automobiles [11]. Extracting specific features to detect lanes in the features extraction process, such as color, edge, geometric, and so on [10]. Several techniques, such as Inverse perspective mapping (IPM)/Perspective Transform, filtering technique, edge detection-based technique, image district extraction, morphological operators, neighborhood searching-based feature points, grayscale, thresholding, and clustering. In addition, heterogenous operators, and sliding window also have been used in the past to reduce the effect of noise and to extract lanes conveniently.

The lane model is then fitted with the line segment detector (LSD) and fitting-based methodologies, including B-spline, quadratic, polynomial, parabola, hyperbola, and least square. Bresenham line voting space (BLVS), vanishing point, waveform, geometric modeling, harmony search (HS) algorithm, contrast limited adaptive histogram equalization (CLAHE), random sample consensus (RANSAC), graph-based, seed fill algorithm, histogram analysis, model predictive control (MPC), a region-based iterative seed method, ant colony optimization, scene understanding physics-enhanced real-time (SUPER) method, nested fusion, and linear regression were used. The Lucas-Kanade approach, Kanade-Lucas-Tomasi (KLT), and Lucas-Kanade optical flow have matched the lane model. Meanwhile, the most extensively used algorithms for tracking road lane detection are the Kalman filter, lane categorization, and parabola equation. Tracking is often employed as a post-processing step to compensate for lighting fluctuations [11]. As a result, tracking aids in incorrectly detecting occlusion due to faulty lane markers [12].

Table V shows the details of the feature extraction, line model fitting, and lane line tracking approaches used in the geometric modeling-based lane detection method. First, feature extraction methods include several techniques such as perspective transform, thresholding, filtering, edge detector, image district extraction, grayscale, clustering, neighborhood searching-based feature points, sliding window, morphological operations, and heterogeneous operators.

Next, Line Model Fitting contains several approaches such as LSD, fitting, BLVS, vanishing point, waveform, geometric analysis, HS algorithm, CLAHE, RANSAC, graph-based, seed fill algorithm, KLT, Histogram analysis, MPC, a region-based iterative seed method, ant colony optimization, SUPER

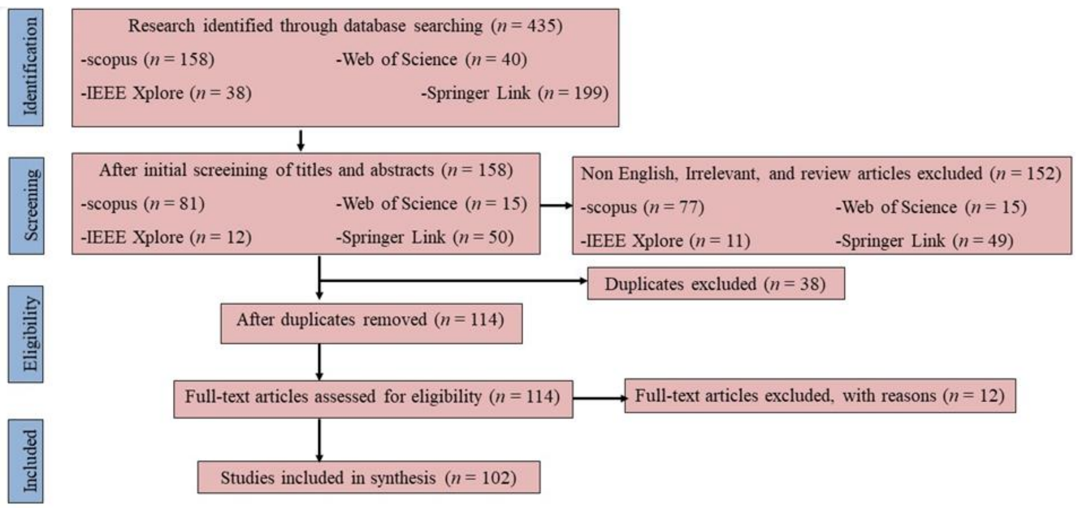

FIGURE 3. Preferred Reporting Items for Systematic Review and Meta-Analysis (PRISMA) Diagram. The research identified through four database searching was 435 publications.

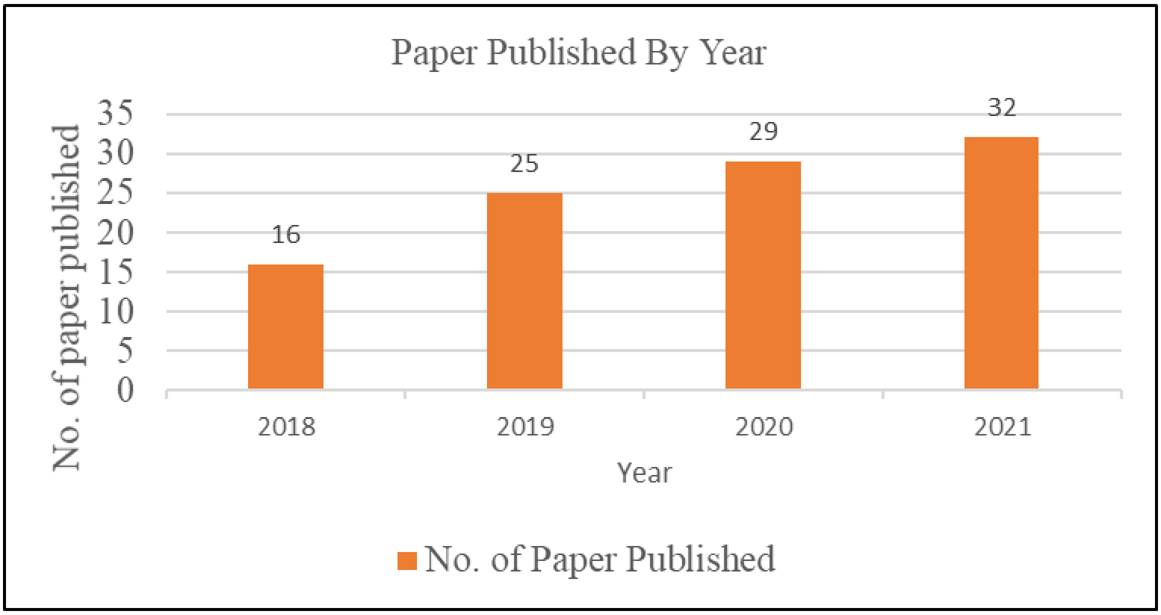

FIGURE 4. Distribution publications for the year 2018-2021. The trend for the statistics of the published papers is increasing every year. The graph show that the lane detection study is still relevant for the upcoming year.

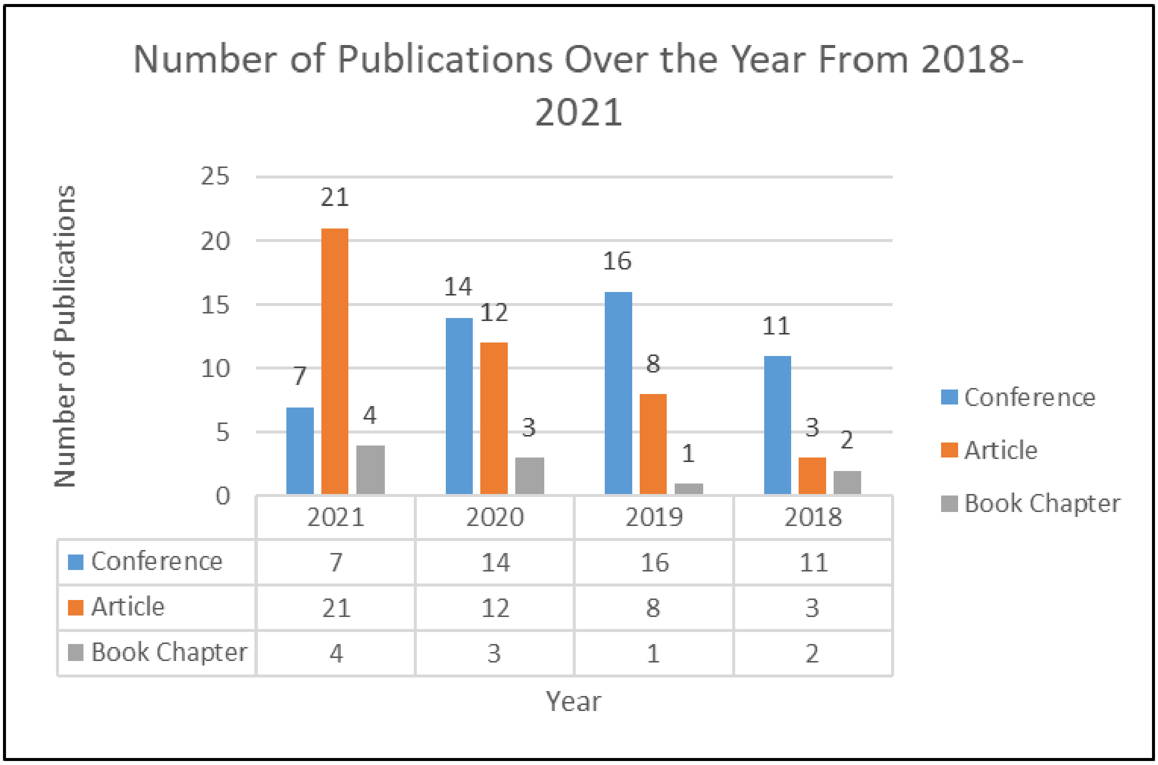

FIGURE 5. The number of publications over the year from 2018-2021. The number of publications for journal articles and book chapters has been increasing over the year. Meanwhile, the conference publications are fluctuating in these four year.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

TABLE I
CHOSEN PUBLICATIONS, SOURCE TITLE, AND THE NUMBER OF CITATIONS

No.	Ref.	Year	Source Title	Cited By
1	[15]	2021	Neurocomputing	-
2	[16]	2021	Journal of Ambient Intelligence and Humanized Computing	11
3	[17]	2021	2nd International Conference for Emerging Technology, INCET 2021	-
4	[18]	2021	Proceedings - International Conference on Artificial Intelligence and Smart Systems, ICAIS 2021	-
5	[19]	2021	Sensors	2
6	[20]	2021	Sensors	3
7	[21]	2021	Journal of Electrical Engineering and Technology	-
8	[22]	2021	Advanced Structured Materials	18
9	[10]	2021	Journal of Advanced Transportation	-
10	[23]	2021	IEEE Access	-
11	[24]	2021	Journal of Supercomputing	-
12	[25]	2021	International Journal of Systems Assurance Engineering and Management	1
13	[26]	2021	Soft Computing	1
14	[27]	2021	International Journal of Advanced Robotic Systems	-
15	[28]	2021	IEEE Transactions on Intelligent Vehicles	4
16	[29]	2021	International Journal of Advanced Robotic Systems	-
17	[30]	2021	Lecture Notes on Data Engineering and Communications Technologies	1
18	[31]	2021	Journal of Electrical Engineering and Technology	-
19	[12]	2021	Complex & Intelligent Systems	17
20	[32]	2021	Cognitive Computation	-
21	[33]	2021	Proceedings of International Conference on Machine Intelligence and Data Science Applications	-
22	[34]	2021	Transactions on Computer Systems and Networks	-
23	[35]	2021	Multimedia Tools and Applications	-
24	[36]	2021	Multimedia Tools and Applications	1
25	[37]	2021	Science China Technological Sciences	1
26	[38]	2021	Computing	1
27	[39]	2021	Applications of Advanced Computing in Systems, Proceedings of International Conference on Advances in Systems, Control and Computing	-
28	[40]	2021	International Conference on P2P, Parallel, Grid, Cloud and Internet Computing	1
29	[41]	2021	International Conference on Intelligent Computing	-
30	[42]	2021	Journal of Ambient Intelligence and Humanized Computing	-
31	[43]	2021	Image and Graphics	-
32	[44]	2021	The 10th International Conference on Computer Engineering and Networks	-
33	[11]	2020	IEEE Transactions on Intelligent Transportation Systems	-
34	[45]	2020	16th IEEE International Conference on Control, Automation, Robotics and Vision, ICARCV 2020	-
35	[46]	2020	Journal of Ambient Intelligence and Humanized Computing	4
36	[47]	2020	Proceedings - IEEE 18th International Conference on Dependable, Autonomic and Secure Computing	-
37	[25]	2020	Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020	-
38	[48]	2020	IEEE International Conference on Electro Information Technology	-
39	[49]	2020	Computers and Electrical Engineering	19
40	[50]	2020	ACM International Conference Proceeding Series	-
41	[51]	2020	Signal Processing: Image Communication	-
42	[52]	2020	Applied Sciences	4

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

43	[53]	2020	International Journal of Automotive Technology	12
44	[54]	2020	International Journal of Semantic Computing	1
45	[55]	2020	International Journal of Image and Data Fusion	-
46	[56]	2020	Proceedings - International Conference on Pattern Recognition	2
47	[57]	2020	Advances in Intelligent Systems and Computing	1
48	[58]	2020	Recent Advances in Computer Science and Communications	11
49	[59]	2020	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)	19
50	[60]	2020	Asian Conference on Pattern Recognition	-
51	[61]	2020	IEEE Access	4
52	[62]	2020	Lecture Notes in Networks and Systems	5
53	[63]	2020	IEEE Transactions on Vehicular Technology	102
54	[64]	2020	Advances in Intelligent Systems and Computing	1
55	[65]	2020	Advances in Intelligent Systems and Computing	-
56	[66]	2020	International Conference on Green Technology and Sustainable Development (GTSD)	-
57	[67]	2020	Evolutionary Intelligence	-
58	[68]	2020	Journal of Intelligent & Robotic Systems	-
59	[69]	2020	Computational Intelligence in Data Science	-
60	[70]	2020	Databases and Information Systems	-
61	[71]	2020	Iberian Robotics conference	1
62	[72]	2019	IEEE International Conference on Robotics and Biomimetic, ROBIO 2019	7
63	[73]	2019	Journal of Visual Communication and Image Representation	5
64	[74]	2019	Proceedings - 2019 Chinese Automation Congress, CAC 2019	-
65	[75]	2019	2019 IEEE International Conference on Electrical, Control and Instrumentation Engineering, ICECIE 2019	1
66	[76]	2019	International Conference on Control, Automation and Systems	6
67	[77]	2019	Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering	5
68	[78]	2019	Proceedings of the 3rd World Conference on Smart Trends in Systems, Security, Sustainability, WorldS4 2019	2
69	[79]	2019	Chinese Control Conference, CCC	3
70	[80]	2019	Proceedings of 2019 International Conference on System Science and Engineering, ICSSE 2019	20
71	[81]	2019	Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition	33
72	[82]	2019	ACM International Conference Proceeding Series	2
73	[83]	2019	SAE Technical Papers	-
74	[84]	2019	Sensors	13
75	[85]	2019	IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference	-
76	[86]	2019	IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences	-
77	[87]	2019	Computing in Science and Engineering	2
78	[88]	2019	Machine Vision and Applications	18
79	[89]	2019	18th International Conference on Advances in ICT for Emerging Regions, ICTer 2018 - Proceedings	2
80	[90]	2019	AmE 2019: Automotive meets Electronics 2019 10th GMM Conference	1
81	[91]	2019	IET Conference Publications	9
82	[92]	2019	International Journal of Web and Grid Services	-
83	[93]	2019	Journal of Institute of Control, Robotics and Systems	-
84	[94]	2019	Advances in Intelligent Systems and Computing	1
85	[95]	2019	Lecture Notes in Computational Vision and Biomechanics	-
86	[96]	2019	International Symposium on Signal Processing and Intelligent Recognition Systems	2

87	[97]	2018	2018 New Generation of CAS, NGCAS 2018	3
88	[98]	2018	Sensors	20
89	[99]	2018	2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2018	21
90	[100]	2018	IEEE Intelligent Vehicles Symposium, Proceedings	1
91	[101]	2018	2018 6th International Conference on Control Engineering and Information Technology, CEIT 2018	-
92	[102]	2018	IET Intelligent Transport Systems	42
93	[103]	2018	Swarm and Evolutionary Computation	37
94	[104]	2018	Asia Life Sciences	14
95	[105]	2018	IFAC-PapersOnLine	2
96	[106]	2018	IEEE International Conference on Computer and Communication Engineering Technology (CCET)	-
97	[107]	2018	Chinese Control Conference, CCC	7
98	[108]	2018	2018 6th International Conference on Control Engineering and Information Technology, CEIT 2018	5
99	[109]	2018	IFIP International Conference on Artificial Intelligence Applications and Innovations	17
100	[22]	2018	International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing	3
101	[110]	2018	Image and Video Technology	8
102	[111]	2018	Computational Science and Technology	-

algorithm, nested fusion, Lucas-Kanade optical flow, and linear regression. Meanwhile, three techniques have been applied for line tracking approaches the Kalman filter, lane classification, and parabola equation.

Geometric modeling/traditional lane detection approaches are used in much literature, such as by D. Kavitha & S. Ravikumar [16]. The input image is first transformed into a greyscale image from a color image. The noise is eliminated, and edge detail enhancement is performed for the image preprocessing procedure phase. After converting to greyscale, the author used the adaptive median filter (AMF) to reduce/remove noise and then used the Laplacian-based technique for contrast enhancement. After the preprocessing stage of the task is completed, the edges in the image are recognized using the Canny operator for the feature extraction stage. The Hough transform is used to fit the line model after the edges have been detected. The Hough transform is commonly used to extract characteristics affecting the geometry of an input image. The lane is then detected using the hyperbola fitting technique. S. Ghanem *et al*. [12] also proposed a geometric modeling-based method for detecting road lanes, including image processing, feature extraction, line fitting model, and lane line tracking pipelines. First, the Region of Interest (ROI) is used in the image processing stage to remove another object unrelated to the lane markers. In the feature extraction step, edges are extracted from the image using the Canny approach, which is robust against noise. Second, the Hough Transform is used to extract the line segments. After that, the input is filtered using the standard deviation (SD) filter. This textural filter aids in the provision of local intensity variation information. When the texture is smoother, the SD filter's response is smaller. As a result, the SD filter is employed in this research to show the degree of pixel value variability in a region. This SD filter computes the SD of the pixels in the vicinity of the pixel of interest. In addition to the SD filter, the Gaussian filter can remove noise. This study uses least-square fitting to fit the line model. Meanwhile, the Kalman filter is used to accomplish the lane tracking procedure in this research since it helps to converge to actual values faster than other methods.

After that, J. Gong *et al*. [35] used the double threshold approach to preprocess the self-collected road image and get the ROI. The region of interest, which includes lane line information, is intercepted to reduce background interference on the road and improve the algorithm's real-time performance. The grey value of the image is then processed utilizing image enhancement employing exponential function transformation. After a nonlinear grey change, the low grey value background area becomes darker, while the lane line area becomes lighter in color. As a result, the contour of the high-grey-valued area becomes more visible, and the contrast improves. The method effectively increases the difference between the lane line region and the background information, lowering the threshold selection difficulty. The image grey value adjustment and image smoothing were carried out only in the significant region of the road to tackle the problems of lane detection taking a long time and having poor noise resistance. The modified Canny operator was then used to extract the lane line edge. When the Otsu threshold was chosen, the Kalman filter technique was used to anticipate the ideal point in the following image series using optimized autoregressive data processing features. The OTSU technique is an approach for determining the image binarization segmentation threshold proposed by Japanese expert OTSU. The high and low thresholds are supposed to be known. According to the OTSU basic principle, the image is separated into three sections: the background part, the suspected foreground fraction, and the foreground part. Following that, a practical multi-layer evaluation function was constructed to implement the online adjustment of lane lines using the straight-line fitted by the Hough Transform. A. Kasmi *et al*. [45] is another paper that proposed the traditional technique. Initially selecting the best Region of In terest, the author used the conventional method for detecting the road lane. Following choosing the most informative ROI, the RANSAC approach detects the segment within the ROI. Finally, to track the road lane, the Kalman filter is used.

Next, B. Akbari *et al*. [19] used the geometric modeling technique, which uses the ROI for preprocessing and the Canny operator to extract the edge feature, and the Hough transform to filter out unwanted edges and lead to straight lines. The vanishing point then filters out the image's irrelev ant straight-line segments. As a result, the B-spline clustering and IPDA filter is also utilized in this literature to detect the road lane efficiently.

TABLE II
PUBLICATIONS OF LANE DETECTION THROUGH JOURNAL

No.	Journal Title	No. of Publications
1	Sensors	5
2	Journal of Ambient Intelligence and Humanized Computing	2
3	International Journal of Advanced Robotic Systems	2
4	Journal of Electrical Engineering and Technology	2
5	Multimedia Tools and Applications	2
6	IEEE Access	2
7	Journal of Supercomputing	1
8	International Journal of Systems Assurance Engineering and Management	1
9	Soft Computing	1
10	Neurocomputing	1
11	IEEE Transactions on Intelligent Vehicles	1
12	Asia Life Sciences	1
13	Swarm and Evolutionary Computation	1
14	Complex & Intelligent Systems	1
15	Cognitive Computation	1
16	Journal of Advanced Transportation	1
17	Science China Technological Sciences	1
18	Computing	1
19	IEEE Transactions on Intelligent Transportation Systems	1
20	Computers and Electrical Engineering	1
21	Signal Processing: Image Communication	1
22	Applied Sciences	1
23	International Journal of Automotive Technology	1
24	International Journal of Image and Data Fusion	1
25	Recent Advances in Computer Science and Communications	1
26	Journal of Institute of Control, Robotics and Systems	1
27	IEEE Transactions on Vehicular Technology	1
28	Evolutionary Intelligence	1
29	Journal of Intelligent & Robotic Systems	1
30	Journal of Visual Communication and Image Representation	1
31	Journal of Automobile Engineering	1
32	IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences	1
33	Computing in Science and Engineering	1
34	Machine Vision and Applications	1
35	International Journal of Web and Grid Services	1

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

TABLE III
PUBLICATIONS OF LANE DETECTION THROUGH CONFERENCE

No.	Conference Title	No. of Publications
1	Advances in Intelligent Systems and Computing	5
2	ACM International Conference Proceeding Series	2
3	2nd International Conference for Emerging Technology, INCET 2021	2
4	Chinese Control Conference, CCC	2
5	IET Conference Publications	2
6	2018 6th International Conference on Control Engineering and Information Technology, CEIT 2018	2
7	Proceedings - International Conference on Artificial Intelligence and Smart Systems, ICAIS 2021	1
8	16th IEEE International Conference on Control, Automation, Robotics and Vision, ICARCV 2020	1
9	Proceedings - IEEE 18th International Conference on Dependable, Autonomic and Secure Computing	1
10	Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020	1
11	IEEE International Conference on Electro Information Technology	1
12	International Conference on Intelligent Computing	1
13	International Journal of Semantic Computing	1
14	Proceedings - International Conference on Pattern Recognition	1
15	The 10th International Conference on Computer Engineering and Networks	1
16	Asian Conference on Pattern Recognition	1
17	International Conference on Green Technology and Sustainable Development (GTSD)	1
18	Iberian Robotics conference	1
19	IEEE International Conference on Robotics and Biomimetic, ROBIO 2019	1
20	Proceedings - 2019 Chinese Automation Congress, CAC 2019	1
21	2019 IEEE International Conference on Electrical, Control and Instrumentation Engineering, ICECIE 2019	1
22	International Conference on Control, Automation and Systems	1
23	Proceedings of the 3rd World Conference on Smart Trends in Systems, Security and Sustainability, WorldS4 2019	1
24	Proceedings of International Conference on Advances in Systems, Control and Computing	1
25	Proceedings of 2019 International Conference on System Science and Engineering, ICSSE 2019	1
26	Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition	1
27	IEEE Intelligent Vehicles Symposium, Proceedings	1
28	IEMECON 2019 - 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference	1
29	18th International Conference on Advances in ICT for Emerging Regions, ICTer 2018 - Proceedings	1
30	AmE 2019: Automotive meets Electronics 2019 10th GMM Conference	1
31	International Conference on P2P, Parallel, Grid, Cloud and Internet Computing	1
32	2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2018	1
33	International Symposium on Signal Processing and Intelligent Recognition Systems	1
34	2018 New Generation of Circuits & Systems CAS Conference, NGCAS 2018	1
35	Proceedings of International Conference on Machine Intelligence and Data Science Applications	1
36	International Federation of Automatic Control-PapersOnLine	1
37	IEEE International Conference on Computer and Communication Engineering Technology (CCET)	1
38	IFIP International Conference on Artificial Intelligence Applications and Innovations	1
39	International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing	1
40	SAE Technical Papers	1
41	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)	1

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

TABLE IV
PUBLICATIONS OF LANE DETECTION THROUGH BOOK CHAPTER

No.	Book Chapter Title	No. of Publications
1	Advanced Structured Materials	1
2	Lecture Notes on Data Engineering and Communications Technologies	1
3	Transactions on Computer Systems and Networks	1
4	Image and Graphics	1
5	Lecture Notes in Networks and Systems	1
6	Computational Intelligence in Data Science	1
7	Databases and Information Systems	1
8	Lecture Notes in Computational Vision and Biomechanics	1
9	Image and Video Technology	1
10	Computational Science and Technology	1

These methods are quick and easy to use but require manual parameters. Furthermore, while they can function well in routine situations, they cannot adjust to changing conditions such as lighting and occlusion [10]. Furthermore, while conventional lane detection methods are frequently quick and straightforward and can meet real-time requirements, the road environment is constantly changing due to weather, light, and cars. The findings are not qualified with high accuracy [15].

2) ARTIFICIAL INTELLIGENCE

Artificial intelligence (AI) is the idea of computers, specifically computer systems, imitating human intelligence processes. Expert systems, natural language processing, speech recognition, and machine vision are examples of AI applications. AI systems generally absorb enormous volumes of labeled training data, analyze it for correlations and patterns, and use them to forecast future states. For example, machine learning and deep learning are the AI algorithms that detect lanes. Unfortunately, most traditional lane detection systems suffer from either processing time that does not meet real-time needs or inefficiency in a complex environment that also fails to meet the total availability restriction of such a core function [46]. The two branches of AI-based methodology described in this paper are machine learning and deep learning-based techniques. However, deep learning has become more popular than machine learning due to its excellent performance in either classification or detection using image frames as input to the network technique.

a: Machine Learning (ML)

Machine learning is a method that combines data and algorithms to mimic the way humans learn and increase its accuracy over time. For example, several lane detection experiments in autonomous vehicles have been conducted. Bayesian Classifier, Haar Cascades, Extreme Learning Machine, Support Vector Machine, and Artificial Neural Network are machine learning techniques employed in this field.

R. R. Dhanashirur [96] proposes a lane detecting framework based on machine learning. In this work, the dataset is initially preprocessed using adaptive thresholding, the Otsu approach to estimating ROI in an image. The Cascaded Dempster Schafer Combination Rule is then used to create a form of Bayesian learning. Finally, outliers are removed from the post-process data using morphological procedures such as erosion and dilation consecutively using a tiny kernel.

Afterward, Z. Feng and Werner Wiesbeek [90] advocated combining machine learning and deep learning. The author handles the lane detection problem by first developing a semantic segmentation-based technique using a 5-layer SegNet segmentation neural network, divided into the encoder and decoder networks. However, based on the segmentation results, there are segmentation uncertainties: areas not belonging to the lane will be segmented into the lane in specific single cycles shortly, and vice versa. As a result, Bayes' theorem can improve the segmentation's stability. The Radial Basis Function (RBF)-kernel and Support Vector Machine are also tested to create a robust model for detecting the road lane.

The detection of sharply curved lanes remains a complex problem. As a result, M. Fakhfakh [46] suggested a unique curved lanes characterization and estimation algorithm based on a Bayesian framework for estimating multi-hyperbola parameters to recognize curved lanes under challenging settings. First, the trajectory over each section is modeled by a hyperbola, whose parameters are computed using the suggested hierarchical Bayesian model. Next, the input image is preprocessed to extract contours, characterizing the extracted lanes by fitting them to the chosen analytical model. Finally, a Bayesian approach is proposed to accurately define the curving lane over the entire image by estimating the hyperparameters of the N hyperboles.

b: Deep Learning (DL)

Due to the advancement of deep learning, numerous strategies have been presented to increase the performance of

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

TABLE V
Feature Extraction, Line Model Fitting and Line Tracking Techniques For Geometric Modelling-Based Method in Lane Detection

Feature Extraction	Line Model Fitting	Line Tracking
Inverse perspective mapping (IPM) / Perspective transform	Line segment detector (LSD)	Kalman filter
Thresholding ➢ Symmetrical local threshold (SLT) ➢ Segmentation threshold ➢ Adaptive threshold ➢ Otsu's threshold Quadratic threshold	Fitting ➢ B-spline curve fitting ➢ Quadratic fitting ➢ Polynomial fitting ➢ Parabola fitting ➢ Hyperbola fitting Least square fitting	Lane classification
Filtering ➢ Gaussian filter ➢ Average filter ➢ Median filter ➢ Fuzzy noise reduction filter (FNRF) ➢ Binary filter ➢ Histogram filter ➢ Steerable filter ➢ Integrated Probabilistic Data Association (IPDA) ➢ Standard deviation filter Colour filter adjustment	Bresenham line voting space (BLVS)	Parabola equation
Edge Detector ➢ Canny edge detector ➢ Sobel edge detector ➢ Adaptive edge detector Filter kernels edge detector	Vanishing point	
Image district extraction	Waveform	
Grayscale	Geometric analysis	
Clustering ➢ Density-based spatial clustering of applications with noise (DBSCAN) ➢ Attentive voting based clustering K-Means clustering	Harmony search (HS) algorithm	
Neighbourhood searching-based feature points	Contrast limited adaptive histogram equalization (CLAHE)	
Sliding window	Random sample consensus (RANSAC)	
Morphological operations	Graph-based	
Heterogeneous operators	Seed fill algorithm	
	Kanade-lucas-tomasi (KLT)	
	Histogram analysis	
	Model predictive control (MPC)	
	A region-based iterative seed method	
	Ant colony optimization	
	Scene understanding physics-enhanced real-time (SUPER) algorithm Nested fusion	
	Lucas-Kanade optical flow	
	Linear regression	

lane detecting tasks using this approach compared to previous methods [15].

Recent improvements in DL architectures have considerably impacted the refinement of derived features for lane detection tasks. Neural networks have handled traditional ROI generation, filtering, and tracking approaches [11]. The Convolutional Neural Network (CNN) is used in the majority of deep learning methods [58], [112]. As CNN has grown in popularity, new concepts and systems have been offered [10].

Furthermore, with its remarkable feature extraction capabilities, Convolutional Neural Network (CNN) has been widely employed in computer vision since AlexNet [113]. As a result, many excellent neural networks have been proposed. Because of its simplicity and modular nature, it has been widely utilized as a backbone network. ResNet variations, such as ResNet [114] and ResNeXt [115], have been released recently. Lane detection is another application of these networks [11]. Other methods for detecting lanes in continuous frames include CNN, Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM) [22]. Also, the Deep Learning method's Fully Convolutional Network (FCN) [6] is commonly used for semantic segmentation, and it has been swiftly adopted in numerous ways [116], [117], as well as lane detection approaches [118], [119]. An encoder-decoder structure [120], as well as an end-to-end architecture, are two network model structures that are frequently employed in many computer vision tasks [15], [121], [122]. Semantic segmentation approaches [123]–[125] are also applied to identify the background and lane pixels. Then, to get lane location [10], instance segmentation methods [126] are utilized.

In recognizing the road lane, the DL adaptation approach can be used in various ways. Several researchers advised employing the DL methodology independently, and others suggested integrating it with another method. Incorporating this network increases the network's efficiency in detecting the lane mark under challenging settings. DL + geometric modeling, DL + ML, and DL + DL are examples of methods that can be combined with another. Aside from that, combining DL with an attention mechanism has recently been presented as a novel means of integrating this technology. This is a new proposed state-of-the-art technique that other researchers can investigate further.

i) Conventional Deep Learning

Several works of literature built a lane detection system using this article's stand-alone deep learning-based technique. For example, Z. Wu *et al*. [29] proposed a convolutional neural network-based method for recognizing lanes in driving video images. The expectation line represents an autonomous vehicle's driving behavior in greater detail. Using the long short-term memory-based approach, the predicted line is then used to estimate the vehicle's future trajectory. Due to prior information, autonomous cars may drive smoothly by combining a convolutional neural network with long short-term memory-based techniques (convLSTM).

Similarly, Y. Sun *et al*. [72] use atrous convolution and spatial pyramid pooling techniques to construct a new network-based deep learning method for lane detection. LaneNet is used to build the network, consisting of one encoder and two decoders. The Embedding Decoder and the Binary Decoder are the names of the two decoders. The author uses a sequential mix of the Atrous ResNet-101 and the Spatial Pyramid Pooling (SPP) networks to replace LaneNet's original encoder. Meanwhile, the Embedding Decoder and Binary Decoder architecture are similar, except for the number of output dimensions. The suggested lane detection system in [78] is based on the Drive Works LaneNet pipeline, which uses camera images. This paper presents an integrated framework for autonomous driving based on the NVidia deep neural network multi-class object identification framework, the lane detection framework, and the free space detection framework. This framework can also be used for localization based on map matching, mapping, and path planning in autonomous driving solutions. Finally, in [81], J. Philion proposes a revolutionary, utterly convolutional lane detection model that learns to decode lane structures instead of depending on post-processing to infer structure.

Meanwhile, E. S. Dawam and X. Feng proposed a computer vision-based road surface marking identification system in [47], serving as an additional layer of data for AVs to choose from. The authors used YOLOv3 in the cloud to train the detector to recognize 25 different road surface markings using over 25,000 images. The experiment results show that the detection accuracy and speed are reasonably good.

Traditional approaches based on handcrafted characteristics are less reliable and computationally expensive due to the lack of distinguishing features and several road occlusions. R. Muthalagu *et al*. [36] proposed stand-alone deep learning to deal with this by learning both the lane markings segmentation and the localization and geometry of each lane in the form of critical points using a compact and efficient multi-stage Convolutional Neural Network (CNN) architecture. The proposed methodology combines a lane mask proposal network with a lane key-point determination network to correctly estimate the key points representing the vehicle lanes' left and correct lane markings. Finally, D. K. Dewangan *et al*. [38] suggested a semantic segmentation architecture encoder-decoder network. A hybrid model based on UNet and ResNet has been adopted in this direction. First, the image was down-sampled, and the required features were identified using ResNet-50 as a segmentation model. Then, UNet was used to up-sample and decode the segments of the images using the detected features.

ii) Deep Learning + Geometric Modelling

Several researchers combine a deep learning-based methodology with geometric modeling methods to increase the efficiency of detecting the road lane. While training on manually labeled data, deep neural networks have demonstrated their potential to reach competing accuracy and time complexity. However, the lack of segmentation masks for host lanes in adverse road environments limits the applicability of fully supervised algorithms to such a situation. To address this issue, R. Yousri *et al*. [23] propose combining classical computer vision techniques and deep learning approaches to establish a reliable benchmarking framework for lane recognition tasks in complicated and dynamic road scenarios.

To begin, researchers tested an automatic segmentation method based on a series of traditional computer vision approaches. This technique generates appropriate weak labels by precisely segmenting the semantic region of the host lane in the complex urban images of the nuScenes dataset utilized in this framework. To begin with, the checkerboard-based calibration technique is used to correct distortion. Then, using the vertical mean distribution (VMD) approach, an adaptive region of interest (AROI) is chosen. Finally, the

IEEE Access
Multidisciplinary ⁞ Rapid Review ⁞ Open Access Journal

author employs the progressive probabilistic Hough transform (PPHT) to locate the lane region and calculate the vanishing point. To limit the undesirable consequences of such off-lane information, filtering must be done by masking areas of the images. As a result, the author segments the road using an adaptive algorithm based on a horizon line. The Canny approach is then used to deal with the arbitrary lane shapes discovered in the photos. Because the lane lines are parallel, straight, and of varying colors, image processing techniques retain and enhance these characteristics. Then, color space conversion and morphological processes ensure precise lane segmentation. The morphological top-hat procedure is commonly employed to separate the image's brighter portions from their darker surroundings. In the photos, bright pixels depict lane lines.

As a result, top-hat operation aids incorrect lane identification in unforeseen lightning variations by denoising and enhancing contrast. After using the perspective transform, line fitting is required to complete the segmentation stage to identify the lane region and improve lane features. Next, a sliding window search is used to iterate over different line shapes for more flexible fitting when dealing with arbitrary forms. Finally, the images are unwrapped to the standard view using the inverse perspective transform, and ground truth labels are constructed using single-channel conversion. SegNet, Modified SegNet, U-Net, ResNet, and ResUNet++ are five state-of-the-art FCN-based architectures trained and benchmarked using the data. The work's contributions include the first time ResUNet++ was introduced on the lane detection task, where it outperformed the other tested models, and the introduction of a robust lane detection using an ensemble-based approach, as well as testing the models by looking at the ensemble prediction of the top three models in shadowy scenes and obscuring road scenarios.

Traditional computer vision (CV) techniques are often time-consuming, require more processing resources, and employ complex algorithms to analyze the lane images' detailed properties. This research [24] proposes a deep convolutional neural network (CNN) architecture that avoids the complexities of existing CV techniques to address this issue. As a result, CNN is considered a viable method for lane marking prediction, although improved performance necessitates hyper-parameter modification. An S-Shaped Binary Butterfly Optimization Algorithm (SBBOA) is used in this paper to improve the initial parameter setting of the CNN. This method chooses the relative CNN parameters for precise lane marking. The suggested SBBOA optimized CNN framework extracts the lane's pixel attributes before using the CNN architecture to predict the lane. In this study, each lane line is considered as a specific circumstance. The SBBOA-CNN classifier determines which pixel belongs to which lane and turns that knowledge into a parameter description.

Next, N. Kanagaraj *et al*. [25] show how to improve the efficiency of autonomous vehicles by using Convolutional Neural Networks with Spatial Transformer Networks and real-time lane detection. First, the pipeline converts a real-time image to grayscale and smoothes the edges with a Gaussian Blur to reduce noise. Applying a Canny function to aid edge detection is the next step in the process. The edges in the image are obtained after performing the Canny process by measuring the gradients of adjacent pixels. A significant change in gradients can identify an edge. Because the lanes will be found in the bottom half of the image, a region of interest is constructed that corresponds to that portion of the image. A Hough transformation is used to obtain the image's lane lines in the next stage. A single long lane line separates the left and right lanes. This is accomplished by filtering the lines based on their slope to determine which lines belong to which range and disregarding the others. The left and right lanes for the region of interest are found this way. The next step is to overlap the lane lines with the original image to combine the images. The camera calibration matrices and distortion coefficients are computed before performing a distortion correction to raw images and creating a threshold binary image using color transform and gradients. After that a perspective transformation creates a bird's-eye view of the image. Even when lane lines in an image are parallel, perspective causes it to appear to converge from a distance.

It is easy to remove the curvature of lane lines from this perspective. The convolution is then used with a sliding window to maximize the number of heated pixels in each window. The Spatial Transformer Network (STN) then interpolates images using a learnable transformation that removes spatial invariance. The STN block enhances the classifier's accuracy when used in a convolutional neural network. Due to input changes, convolutional neural networks might suffer from a lack of robustness. Scale, viewpoint, and backdrop clutter are examples of these variances. The STN aids in the reduction of these difficulties brought on by input variability. Because of its versatility, an STN can be introduced into any model area. They can also be trained using only one backpropagation algorithm.

H. Zhan and L. Chen [74] suggested a lane line detection technique based on image processing and deep learning based on the FPGA development platform to accomplish the fast lane line detection effect of structured roadways, with speeds up to 104 FPS. First, the camera captures road data, which is then transferred to the FPGA as image data via the AXI protocol. This part aims to convert data into RGB24 format, including data format conversion and transmission interface conversion. The image from the camera is first subjected to data preprocessing, which provides for data format conversion and transfer interface conversion. In addition, an image processing approach that includes threshold segmentation, inverse perspective transformation, and lane line quadratic curve fitting is used to detect lane lines. The final output detection results are the curvature radius of the present lane, the lane's bending direction, the path and distance of the vehicle deviating from the lane center, and so on. At the same time, the lane line coordinates are provided to enable the lane line type identification module to intercept the identification area dynamically. As a result, this study uses the deep learning (CNN) method to detect lane markers and display the output image.

The authors of [102] present a new lane marking detection system based on lane structure analysis and convolutional neural networks (CNNs). The pavement that serves as the background for the lane markers is first removed in a preprocessing stage. Following that, a region of interest is created using a set of local waveforms from local images, and a CNN classifier is used to find lane marking candidates. Finally, the lane geometry analysis stage determines whether the item is a lane marking. A map relative localization method based on road lane matching [50] is developed. When GNSS data is neither exact nor unavailable, the technique provides lane-level location accuracy for autonomous vehicle driving. As a lane detector, the DarkSCNN neural network was deployed. The inverse perspective transforms processes the detection and fits it to the polynomial.

Meanwhile, the Modified Iterative Closest Point algorithm compares two-point clouds: one created using HD-map data and the other using camera data. Furthermore, in [80], images from a front-view camera are captured and fed into a semantic segmentation network to extract features for detecting road lane markings. The network is first built using the U-Net architecture, a convolutional neural network designed for biomedical image segmentation. The Hough Transform method is then used to determine the segmentation network's output lines. Unfortunately, Hough Transform also produces a lot of lines from segmented images. As a result, the K-means Clustering technique is investigated to compute and identify the best line for each road lane marking.

Then, using a combination of semantic segmentation and optical flow estimation networks, S. Lu *et al*. [20] proposed a fast and reliable lane detecting approach. The study was divided into lane segmentation, lane discrimination, and mapping. First, a robust semantic segmentation network was developed for keyframe segmentation, and a fast and slim optical flow estimation network was employed to track non-key frames in lane segmentation. The density-based spatial clustering of applications with noise (DBSCAN) was used to identify lanes in the second part. Finally, a mapping approach for translating lane pixels from the pixel coordinate system to the camera coordinate system and modeling lane curves in the camera coordinate system is proposed, providing feedback for autonomous driving.

First, the preprocessing of input frames in [76] involves removing most of the sky region and performing the automobile dashboard. The frame is then scaled to a resolution of 360x480. This frame is then input into the lane marking segmentation network, which segments out the visible lane marking pixels before using graph-based algorithms to detect instances of segmented lane markings.

The instance segmented output is subjected to perspective transformation (bird's eye view), followed by an attentive voting-based clustering approach and polynomial curve fitting, which yields the final result. Finally, the author created a lane segmentation network with stride convolutions and stride deconvolutions with relu activation in hidden units using the deep learning method, a CNN-based methodology. The research [109] developed a Spatio-temporal, deep learning-based lane boundary recognition approach that can detect lane boundaries accurately in real-time under complex weather circumstances and traffic scenarios. The algorithm is divided into three parts: first, perform the inverse perspective transform and lane boundary position estimation using lane boundaries' spatial and temporal constraints; second, classify the boundary type and regress the lane boundary position using convolutional neural networks (CNN). Finally, the author optimizes the CNN output and uses Catmull-Rom (CR) spline fitting to conduct lane fitting.

Then, in [66], a comprehensive method for detecting lanes and impediments on the road is proposed. A combination of deep learning and a traditional image processing framework was developed for detecting lanes. When the DL approach and the conventional method are combined, data collection time and effort are reduced while performance is maintained. The author first proposed the LiteSeg network architecture. The acquired RGB image is the network's input, and the output is a lane segmentation map with two classes: lane and non-lane. MobileNetV2 is the backbone network with a depth-wise and inverted residual structure. However, the LiteSeg network, which uses the MobileNetV2 backbone, cannot detect all lanes correctly. Because the acquired data contain a lot of noise and fragmentation, the author offers a Hough transform-based lane detection method to fix the problem. In addition, the author creates a lane model using a quadratic polynomial to deal with curvy lanes. After that, the resulting candidate segments are fitted into the lane model using Polynomial curve fitting. The road ROI is then determined using the obtained outermost lanes. After that, the defined ROI will be forwarded to the depth processing task to be processed further.

Finally, the literature in [106] introduced the model pipeline, which consists of three modules: binary semantic segmentation, clustering, and curve fitting. The semantic segmentation module analyzes pixels in an image to see if they belong to a lane line or the background. The clustering module clusters the lane points to form different lane line instances. When the instance segmentation is completed the perspective transformation converts the image into a bird's-eye view. Finally, a curve fitting technique precisely identifies each lane line. To ensure excellent temporal efficiency, the author uses MobileNet as the backbone of CNN in the semantic segmentation module. Furthermore, MobileNet is a valuable model for mobile and embedded vision applications since it uses depth-wise separable convolution. In addition, the author clusters points that correspond to various lane lines using the K-Means clustering algorithm.

iii) Deep Learning + Machine Learning

A machine learning-based strategy is also chosen to integrate with DL to boost the efficiency of lane detection tasks and combine DL with the old method. Lane detection utilizing road features-based algorithms and color feature-based algorithms, according to G. Zhang *et al*. [51], cannot achieve satisfactory performance due to several constraints. For example, the number of lanes is frequently not set, and techniques for detecting lanes are sometimes erroneous. Furthermore, Hough transform-based algorithms interpret straight lines as lanes, leading to street lamps being mistaken for lanes. Similarly, adverse weather, such as rain, will impact lane detecting. Likewise, inadequate lighting and a night setting will produce poor results. However, there are yet no practical solutions for dealing with such issues. As a result, standard approaches are ineffective in detecting lanes in complex traffic situations. In addition, lane detection should be done in real-time. Most algorithms, however, fail miserably at this goal. As a result, by modeling the sophisticated traffic situation, this literature provides a quality-guided lane recognition algorithm that can successfully manage various lanes. The author first uses chessboard images for camera calibration to determine the correspondence between the real-world and image coordinate systems. They then use prior knowledge and picture quality scores to capture image regions of interest that only include lane information. After that, they create a two-stage CNN architecture for lane detection that uses a binary lane mask for lane matching. The author then created a multimodel feature fusion approach for training an SVM to classify image regions. From the lane and non-lane areas, the author created a 137-D multimodel feature by combining a 128-D histogram of gradient (HOG) and a 9-D color moment. They then train an SVM to classify various locations. Next, they use a sliding window approach to build a set of additional regions from the image and SVM to select lane regions for testing. Finally, using image segmentation, they train an SVM to split the image into lane-information sections and non-lane information regions.

Afterward, Z. Feng *et al*. [90] combine DL and ML for lane detection. Deep learning (5-Layer SegNet)-based approach is used first to detect the lane. However, as the segmentation results show, there are segmentation uncertainties as to which areas not belonging to the lane will be divided into the lane in specific single cycles and vice versa. Therefore, Bayes' theorem is used to make the segmentation more stable. As a result, an RBF-kernel SVM (Support Vector Machine) is also tested.

iv) Two Serial Deep Learning

Traditional techniques have yielded significant results but have limitations: (1) lane awareness is challenged by varying weather conditions and illumination. Furthermore, previous methods lack a unifying framework for describing various scenes and (2) the inefficiency of using photos owing to potential label noise. J. Liu [73] introduced a lane detection framework for autonomous vehicles based on learning a comprehensive reference quality-aware discriminative gradient deep model, which uses two types of deep networks. To detect the presence of a lane, the author first creates a gradient-guided deep convolutional network because the gradient value of the lane edge is greater than that of other regions. Then use the entire reference image quality assessment (FR-IQA) method to find more discriminative gradient signals while also utilizing geometric characteristics. Following that, a recurrent neural layer reflects the spatial distribution of identified lanes using difficult-to-define visual cues. Finally, the noisy features are abandoned using the sparsity penalty, and only a small percentage of the tagged images are used in this paper. Next, Q. Zou *et al*. [127] propose a deep hybrid architecture that combines the convolutional neural network (CNN) with the recurrent neural network for lane detection using the same strategy (RNN). A CNN block abstracts information from each frame. The CNN features of several continuous frames with time-series properties are subsequently sent into the RNN block for feature learning and lane prediction.

R. Pihlank and A. Riid [70] introduced a novel neural network-based method that integrates autoencoder structural components, residual neural networks, and densely linked neural networks. The proposed architecture consists of three identically structured connected neural networks that combine the architectures of symmetrical AE (with dimension reducing encoder and expanding decoder), ResNet, and DenseNet, with feature map concatenation providing shortcut connections between encoder and decoder layers. Z. M. Chng *et al*. presented two state-of-the-art algorithms, SCNN + RONELD and ENet-SAD + RONELD, in [56]. Furthermore, as this research indicates, convolutional neural networks (CNNs) are used to train deep learning models in recent state-of-the-art lane detecting algorithms. While these models perform admirably on train and test inputs, they perform poorly on unknown datasets from various contexts. This study proposes a real-time resilient neural network improvement for active lane detection (RONELD), using deep learning probability map outputs to identify, track, and optimize active lanes. They adaptively extract lane points from probability map outputs, detect curved and straight lines, and then use weighted least-squares linear regression on straight lanes to correct fractured lane edges caused by edge map fragmentation in real images. Finally, by tracking previous frames, the author hypothesizes genuine active lanes. Finally, F. Pizzati *et al*. [59] proposed an end-to-end system based on two cascaded neural networks that run in real-time for lane boundary identification, clustering, and classification. They train a CNN for lane boundary instance segmentation as a first step. Then, they extract a description for each observed lane boundary and run it through a second CNN. Instead of lane markings, CNN has been trained to recognize lane boundaries. Then, instead of semantic segmentation, they use instance segmentation on lane boundaries. Mask R-CNN, for example, is a cutting-edge network segmentation technique. ERFNet was also chosen as their baseline model. As a result, this paper uses another CNN to classify each lane boundary, linking the identified boundaries with the ground truth. Furthermore, the architecture for this work is based on H-Net.

v) Deep Learning with Attention Mechanism

In the past, state-of-the-art lane detecting algorithms have outperformed traditional methods in complex scenarios, but they also have limitations. For instance, only a certain number of lanes can be spotted, and the cost of detection time is sometimes prohibitive. Human vision's attention mechanism and methods make network learning more concerning. L. Zhang *et al*. [9] presented a real-time lane recognition system based on an attention strategy to address this issue. The proposed network comprises an encoder module that extracts lanes' features and two decoder modules, a binary decoder and an embeddable decoder, that forecast lanes' instance feature maps. The author employs biologically inspired attention in the encoder to extract features holding a wealth of information about the target area. A correlation between the characteristics produced through convolutions and those extracted by attention is developed to learn the contextual information. The contextual information is combined with features from up-sampling in the decoder to compensate for the lost detailed information. The binary decoder assigns each pixel to two categories: lane or backdrop. The distinct lanes are obtained by using an embeddable decoder. The binary decoder's outputs are then used as one of the inputs to the embeddable decoder, which directs the production of exact pixel points on the lanes.

J. Li *et al*. developed a unique Lane-DeepLab model for high-definition maps [15]. Two new features are included in the suggested method: 1) It optimizes the encoder structure by adding an attention module to the ASPP module; 2) It uses the SEB to merge high-level and low-level semantic information to obtain more great features. Furthermore, in complicated scenarios with changeable weather, the proposed model employs the attention mechanism and contextual semantics to fuse information to determine the lane line for the environment.

F. Munir *et al*. [11] combine the deep learning-based algorithm with the attention mechanism to detect the road lane. Lane detection with a dynamic vision sensor (LDNet) is suggested in this paper, which is constructed as an encoder-decoder with an atrous spatial pyramid pooling block followed by an attention-guided decoder for predicting and decreasing false predictions in lane detection tasks. There is no need for a post-processing step with this decoder. The authors suggested LDNet, a novel encoder-decoder architecture for detecting lane marking using detailed event camera images. LDNet simplifies full-resolution detections by extracting higher-dimensional features from an image. The authors also added an ASPP block to the network's core, which increases the feature map's appropriate field size without increasing the number of training parameters. Additionally, adopting an attention-guided decoder increases feature localization in the feature map, obviating post-processing requirements.

Furthermore, lane detection is essential in advanced driver assistance and autonomous driving systems. However, lane detection is affected by various conditions, including some problematic traffic scenarios. The ability to detect multiple lanes is also critical. R. Zhang *et al*. [10] presented RS-Lane, a lane recognition method based on instance segmentation, to address these issues. This approach is built on LaneNet and takes advantage of ResNeSt's Split Attention to increase feature representation on slender and sparse annotations such as lane markings. Self-Attention Distillation is used in this paper to improve the network's feature representation capabilities without adding inference time. The input photos can be correctly processed in the preprocessing module, making it easier to extract features later. The driving image and associated annotation are translated to a standard format used by the model. The annotated data are utilized for training the network to achieve lane segmentation in the model training step. Denoising and fitting are used in the post-processing stage to obtain the final results from the model's output. The network employs the encoder-decoder framework to conduct semantic and instance segmentation simultaneously, as proposed by LaneNet. The encoder's backbone is ResNeSt, which presents a Split-Attention mechanism. As a result, the authors add two more DAS lines to the network to improve its feature extraction capabilities. SAD allows a network to learn from itself without external data. The lower layers can learn the higher feature representation by mimicking the attention maps of the higher layers. Because the lower layers' ability to represent features increases, the higher layers, and the entire network benefit.

As a result, the decoder executes a deconvolution operation to decode the encoder's feature maps and performs upsampling and classification. The decoder has five levels that correspond to the encoder's layers one-to-one. The author used Unet's skip-connect approach to concatenate the encoder and decoder outputs to make the most of the global context information. There are two branches in the decoder's final layer: binary branch and embedding branch. This study generates the binary branch and embedding branch outputs using two convolutional layers with a 1×1 kernel. The binary branch produces semantic segmentation. The embedding branch makes a three-channel map, meaning each pixel has a 3D embedding vector. The segmentation map output is utilized as a mask, and the mask is applied to the embedding map to generate only the lane pixels embedding the map. The author then applies mean-shift clustering to produce clusters for each lane and the actual outcome of instance segmentation. As a result, the lane model is fitted using cubic spline interpolation.

B. WHAT EQUIPMENT IS BEING USED TO COLLECT THE DATASET FOR THE TRAINING PROCESS?

The input data is the most critical aspect for detecting the road lane. Moreover, dataset preparation is essential for the AI approach, especially during training. As a result of the great dataset preparation in the network model, autonomous cars can manage behavior and make judgments. After reviewing the journal, paper conferences, and a few book chapters, numerous works of literature contained self-collection of data and were also done online. In addition, some researchers compile their dataset for AI training only, then compare it to a publicly available benchmark dataset. On the other hand, several researchers only use self-collect data for training and validation. Meanwhile, several researchers have relied only on the public dataset for training and validation. In road lane marking, radio detection and ranging (radar), a camera, a global positioning system (GPS), and light detection and range (LiDAR) have all been used for the self-collect dataset [23]. Other than that, there are also data from the online simulator collected in various works of literature.

This subsection will describe the details of equipment implementation for self-collect data in lane detecting. Figure 7 depicts the year-by-year publication of papers for data collection equipment from 2018 to 2021. For example, in 2018, 13 published articles used cameras, and one published paper used a simulator for data collection. Next, in 2019, 15 published papers used cameras, and one published paper used a simulator and radar for data collection, respectively. Furthermore, in 2020, about 12 published articles used the camera to collect the dataset. Meanwhile, one paper published utilized lidar, OpenStreetMap, and HD map to collect datasets, respectively. Finally, by 2021, about 13 articles used a camera, and one paper used an HD map to acquire the data set.

1) CAMERA

To begin, the camera can be used to extract road markings. As a result, various cameras have been used, including webcams, Wi-Fi sports camera sensors, Kinect cameras, smartphone cameras, monocular cameras, and stereo vision cameras. Monocular cameras are a cost-effective choice; however, they don't provide depth information. On the other hand, stereo vision cameras allow for the inference of depth information and hence the reconstruction of three-dimensional scenarios for increased functionality, such as collision detection [19]. Furthermore, the reliability and ability of cameras to record every circumstance of the road environment in any direction have recently been enhanced [23]. Therefore, vision sensors are also becoming more effective and less expensive due to current deep learning algorithms [38]. However, despite the prevalence of camera sensors, deep learning algorithms offer a high degree of generalization and learn the crucial elements of the driving environment across multiple layers.

According to the literature, most researchers utilize a camera to detect lane markings. The literature recommended using the camera to self-collect data: B. S. Khan *et al*. [111] used the camera to acquire data. The road image was recorded with a single camera sensor to detect the road marking on the vehicle's front side. As a result, a smartphone camera was placed on the front windshield of the experimental car. The datasets used in this study were from videos captured with a Samsung Galaxy Alpha smartphone (SM-G850F). The image was captured at 30 frames per second mode without video stabilization and had a 1920 x 1080 (.mp4) pixels resolution. The total number of videos applied in the experiment is 15, with 22,500 photos retrieved from them. The images were taken under various imaging situations, including lighting, traffic, and climate. The host vehicle was driven according to the two-second safety guideline during data collection. Maintaining a safe following distance is critical when driving a car, and autonomous driving requires that distance to be established. As a result, the two-second safety guideline criterion is utilized to verify a safer following distance at any speed. According to the rule, any vehicle in front of the driver's car should be kept at least two seconds behind the driver's vehicle. Therefore, about 22500 images of roads were taken at various times of the day and night, with varying lighting and occlusions such as shadows, intricate backgrounds, traffic, light rain, rains, and snow. Images with an after-rain effect can also be obtained. The dataset was taken with a camera installed on the dashboard, and the data gathering took place in Selangor and Kuala Lumpur. The remaining images in the dataset (light rain, rain, after rain, snow) were collected from the internet. They were recorded throughout the day and night under various lighting conditions obstructions and consisted of reflection effect complicated background.

Next, Liu *et al*. [54] deliberately chose roads with shadows, tire skid tracks, and noise. Around Lafayette, Indiana, the author filmed local roads and Interstate Highway 65. Each video clip is about 15 seconds long, allowing the images captured to focus on the desired road features. The video was segmented once the data was collected, and the images were extracted every six frames. In the end, 23,088 useful photo bits were gathered. K. C. Bhupathi and Hasan Ferdowsi [48] also use a camera to capture videos. Utilizing the multiple sliding window method, the accuracy of lane detection is assessed on four video sequences. The camera's position should be fixed and usually expected to be in the vehicle's center. Next, a Toyota Prius autonomous driving research prototype vehicle with Nvidia Drive PX 2 and a

Sekonix GMSL Camera was used by N. Kemsaram and A. Das [78]. In a car, A GMSL connector connects a Sekonix GMSL Camera to a Drive PX 2. Drive a vehicle that has the PX 2 in the trunk. The Sekonix GMSL camera is mounted near the rear-view mirror, behind the front windshield. The data set includes multi-frame images sampled from the driving video.

The video has a frame rate of a vertical resolution of 720 pixels and a width resolution of 1280 pixels with 30 frames per second. Next, the images are dissected and evaluated. However, the quality of several pictures is poor due to the lighting and brightness. This emphasizes the significance of lane prediction. Therefore, the training image sample rate is quite significant. The continuous visuals may be highly similar if the pace is high, rendering the model meaningless. As a result, just one image out of ten is chosen for the training dataset. Therefore, the training data set should increase the lane detection model's identification performance. In addition, the training set should include more images of the curving lane. To begin, more images with curving lanes are extracted from the video. Then, the images with the least pixels are chosen. These images are also altered to create new ones.

The authors then employed a random sample of Zibo city road datasets consisting of three road scenarios: shadow occlusion, lane line wear, and bright illumination [35]. The visual data set in every road situation is collected in the video, which contains about 800 images of typical road scene graphs selected from the collected footage. In addition, 2400 graphs are used in computer simulation investigations. The frame had a resolution size of 512 × 682 pixels. As a result of the camera specifications, all of the original images in the experiment are greyed out. To reduce the vehicle's hindrance on the camera view.

[101] uses a camera positioned 21.5cm above the center of the rear axle and 27cm in front. The test data is acquired while the automobile is driven manually to follow the track's lane. Although the data is captured at 60 frames per second (fps) using the test platform's onboard camera, the evaluations are performed offline to ensure a fair comparison. Finally, in [107], the author employed video sequences with 1225 frames with a resolution of 640 × 480 pixels of complex metropolitan streets, which incorporate difficult traffic situations such as diverse pavement types, passing cars, faded writings, and numerous shallows. In addition, after rain, the author collected a new dataset to test the robustness in various climates. There are 1706 frames in total in these databases.

A Mobileye camera vision sensor was placed ahead of the window shield in [128], and it had a variable updating rate of 50 to 130 milliseconds. The yaw rate sensor, which was mounted near the vehicle's center of gravity and updated every ten milliseconds, was used. Each wheel had its speed sensor updated simultaneously with the yaw rate sensor. A Micro AutoBox DS1501 additionally controlled the car from dSPACE Inc., which used the controller area network (CAN) bus to log data from each sensor. The dataset was collected by Lee and J. H. Moon [100] using the self-driving automobile 'Tu Lian,' as shown in Figure 6. POINT GREY BFLY-PGE-23S6C-C camera sensor was mounted in the 'Tu Lian' for data collection. The focal length of the mounted camera is 2246 millimeters, and the camera was calibrated first.

FIGURE 6. Self-driving car 'Tu Lian' with the camera mounted in the vehicle for data collection. POINT GREY BFLY-PGE-23S6C-C camera sensor was mounted in the 'Tu Lian' for data collection.

The in-vehicle camera collects information about the road environment J. Xiao *et al*. [110]. The road views are acquired using a Basler pia1900-32 gm/gc industrial camera, and the system is based on monocular vision. This lens has an 8 mm focal length. The maximum frame rate for a picture is 32 frames per second. The image has a 1,920 × 1,080-pixel resolution. The images are transferred from the camera to the computer through a Gigabit Ethernet interface. The road video data was collected in Erdos, Inner Mongolia, to verify the proposed algorithm's effectiveness and robustness. These data include tree shadows, pedestrians, vehicle trespass, extreme shadow and light, curves, etc.

Next, H. Zhan and L. Chen [74] used a camera to acquire the data, which they then fed into an FPGA as image data using the AXI protocol. Next, the detection outcome for the real dataset collected from the author's autonomous vehicle was published in [15]. The visual images were chosen from a video with a 3 km duration consisting of road lines, road signs, zebra lines, and double solid lines. The proposed approaches are then tested using a dataset of 314 Estonian orthoframe photos highways with a resolution of 4096 × 4096 pixels. In literature [70], an image segment training and validation dataset is built using 249 of the 314 preprocessed images. There are 36497 image segments in this training/validation dataset, each with a size of 224 × 224 pixels. Next, the images [94] were obtained from open roadways and were 960 × 540 pixels in size. To recognize lane features from actual road images, 6000 images were collected, comprising 2000 images each for straight, curved, and lane change sections. In addition, in [88], the author used a dataset of sceneries from the roadways located at KAIST in Daejeon, South Korea. VSTC-V200G camera installed on a car is used to collect the dataset. The video comprises 640 × 360 pixels of resolution at 20 frames per second with 4335 images.

Several works of literature use a different camera sensor than a standard camera. For example, S. Lu *et al*. [20] self-collected the data set to validate the presented lane detection model. The self-collected data set came from a cheap and average webcam with noticeable occlusion, blurring, and poor illumination in its images. The author gathered over 6000 images, which included varied real-world traffic situations. The dataset for the lane detection challenge is collected using the Kinect camera and the webcam camera. The author of [66] used a Kinect camera installed in a 1:7 RC car to evaluate the system's performance in a tiny driving environment. The dataset contains 1000 labeled images and numerous complex examples to test the algorithm on.

The Wi-Fi sports camera sensor is used in [16] to track the entire route taken by the AV. This Wi-Fi-enabled camera sensor transmits actual video to a smartphone for monitoring. The collected footage is sent to smartphones and cloud storage servers for additional processing through a radio transmitter. A computer vision-based algorithm performs the analysis. The real-time data collection aids with vehicle security and lane detection. Furthermore, the system can improve the functionality of this task by utilizing a Wi-Fi sports type of camera. The employed camera sensor contains a 2 inches screen size and a resolution of the optical sensor of approximately about 12 megapixels. The zoom range on this camera is reasonably priced, and it supports High Definition (HD) video. Furthermore, this camera is simple to set up and use. It can also be connected to a smartphone to track the car. However, for non-volatility, availability, and accessibility, this technique sends the data to cloud storage. Finally, a computer vision technique is applied to process the collected data to identify lanes.

On the other hand, lane detection can be performed using infrared sensors. It is vital to capture live traffic data to detect the road lane, which is why the Wi-Fi sports camera is used. Many video frames are involved because the data is in the form of video, and each video frame must be processed before the vehicle may be warned.

Next, Y. Y. Moon *et al*. [103] collected the video images for the tests with a resolution of 640 x 360 since numerous video clips contain 24 or 29 frames per second (fps). As a result, the execution time for each frame must be less than 1/24 s (14 41.7 milliseconds) or 1/29 second (1434.5 ms). In addition, the images of various road circumstances, such as evening conditions, many noises present, and situations in a tunnel, are used in this work. Finally, in [75], road images are collected with an iPhone at a frame rate of 30fps and 1334 × 750 resolution, with the camera sensor installed on the rear mirror. The area in front of the vehicle is depicted in this image, including trees, a road, cars, pedestrians, and a side view.

2) LIDAR

There are two primary benefits of using the camera. First, this sensor delivers extensive surroundings and is currently the cheapest and most dependable modality for automotive applications. However, this sensor is sensitive to light levels, necessitating a filtering step. LiDAR sensors can be used to solve this problem. For example, regardless of the lighting circumstances, it is practicable to detect whether a LiDAR beam has intercepted asphalt or road painting [129]. This is especially useful when dealing with shadows and darkness, which cameras have trouble handling. Furthermore, LiDAR provides a centimeter-accurate three-dimensional picture of the world. LiDAR, on the other hand, it's more costly than cameras. Nonetheless, advancements in optical technology and rising demand will lower the price of LiDAR.

3) SIMULATOR

Little research in lane detection uses simulators to collect data for training and validation. For example, L. Tran and M. Le [130] used a dataset of around 4000 training images to train a segmentation model for 20 hours, with 2000 images annotated. The information comes from the CARLA simulator. Besides that, the training dataset for Unity3D simulation is then collected by M. C. Olgun *et al*.[108]. A setting was built that resembled the author's real-life roads. An AI controller language in an automobile allows it to appropriately follow waypoints between lanes in a given scenario. Frames representing the car's maneuver are saved in a jpg file; meanwhile, image routes, speed, and steering information are kept in CSV format. This dataset's loss value is more consistent than the manually collected dataset. The lane tracking training dataset contains 12531 authentic images supplemented with 20000 images. Next, in [52], the author employed a pioneer robot vehicle to mimic two different track settings. The program finds the lane using this visual input from the Gazebo simulator. Based on lane identification findings and Matlab output, it calculates the vehicle's angular and linear velocity.

4) RADAR

A high-resolution automotive radar prototype is utilized to collect data in [12-13], [90]. The modulation mode of this radar sensor is FMCW (Frequency Modulated Continuous Wave). The baseband signal can calculate range, relative radial velocity, object angle, and reflection magnitude. The signal processing chain begins with a 2-dimensional FFT (Fast Fourier Transform), CFAR (Constant False Alarm Rate), peak detection, and the maximum likelihood angle estimation technique. The axis of the estimated azimuth angle is evenly spaced. The detecting sites' positions and the object's range will fit into a fan-shaped grid-like pattern.

5) HD MAP

The dataset for lane recognition from HD maps is self-collected in several research. As a navigation backend, all commercial autonomous vehicles use accurate high-definition maps with lane markings. However, the majority

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

of high-definition maps are currently produced manually. The generation of high-definition maps for autonomous driving using auto-assisted multi-category lane recognition [15]. The HD map is defined as a map that consists of the precise coordinates of road lanes in the Universal Transverse Mercator (UTM) coordinate system, as described in [50]. Other elements such as road signs and traffic lights are included, but only road lanes are used in this publication. When a new camera frame is received, the author queries all lanes from the HD map within a given radius of the most recent position estimation. This study, for example, used a distance of 20 meters. Because lane line detection takes time, the author should employ the stance when the camera is triggered. The road lane matching module uses information from the front camera to detect lanes and a slice of an HD map near the most recent localization estimate as input. The module determines the best modification for aligning camera lanes to the HD map with the slightest error. The algorithm utilized is the Iterative Closest Point algorithm.

6) OPENSTREETMAP

OSM datasets have been employed in intelligent transportation systems for various purposes, including road-level localization [131] and lane-level determination [132]. Road detection utilizing images obtained from a camera relies on road priors and contextual information. First, the road backbone is built using an OSM map based on the number of lanes and lane width. The image is then projected with this road geometry, considering the uncertainty associated with the ego-vehicle stance. Finally, before the detection of a lane, the result is used.

The study in [133] uses OSM data before creating a more precise map. After that, the authors provide OSM data and proprioceptive sensor fusion architecture. In the meantime, a similar approach derived from OSM was used to identify ego-lane marking in LiDAR point clouds [45]. Nodes, Ways, and Relations [45] are the three crucial components of OSM data. Nodes are the geometrical elements that represent GPS positions. For example, the roadways network is defined by byways, a detailed list of nodes. As a result, each way (road) is made up of segments [131]. In other words, being a part of a segment is similar to being a part of an OSM Way. As a result, the map matching problem should be recast as matching a GPS point to a segment. As a result, the author employs the map-matching technique described in [131] to select the best path (road). However, as discussed in this literature, the OSM data lacks precise information.

C. WHAT WAS THE DATASET USED FOR THE NETWORK TRAINING, VALIDATION, AND TESTING?

TuSimple [76], KITTI, Caltech, Cityscapes, ApolloScape, and CULane datasets are online road scene datasets or benchmarks that provide training data for various uses. In this section, several popular public datasets will be discussed. The network must be given a meaningful dataset to operate efficiently [108].

1) TUSIMPLE DATASET

The TuSimple dataset is a publicly available traffic-detection data set (light traffic and clear lane markings). Its label for the training dataset consists of continuous lane curves that start at the bottom of the input image and continue until the horizon passes over the vehicles [76]. It consists of large datasets with training, and the testing number is 326 and 2782 in both bad and excellent weather conditions, respectively [36]. They are recorded at various times of the day on two road lanes, three road lanes, and four road lanes or extra highway roadways. The resolution of these RGB input images is 1280 × 720 pixels. Each image also includes the 19 frames with the unlabeled dataset. The annotations are JSON format, showing the lanes' x-position at different discretized y-positions. The literature that used the TuSimple dataset for training or validation has been discussed in this section. In their research, Y. Sun *et al*. [72] utilize this public lane detection dataset. The author generates ground truth instance segmentation maps by drawing lines along with the pixel coordinates of each lane. The lines have a thickness of 15 pixels.

In addition, different labels are assigned to various lanes. The author divided the dataset into three parts: a train set with 3268 images, a validation set with 358 images, and a test set with 2782 images, respectively. Next, the TuSimple dataset was utilized by S. Lu *et al*. [20] to validate the proposed lane detection model. The dataset employed in this study has good visual clarity, no blur, and a low detection difficulty. Besides that, the TuSimple dataset is also used in experiments carried out in [24]. In this study, the TuSimple dataset contains almost 7000 video segments. Each video clip comprises twenty frames in total. Seventy percent of the videos are used for learning in the network, twenty percent for validation, and 105 for testing. In detail, the training, validation, and testing sets contain 4900, 1400, and 700 video clips, respectively. The sample of TuSimple datasets images was taken in a variety of climatic factors. Next, F. Pizzati *et al*. [59] used this dataset, which consisted of 6408 images with a resolution of 1280 × 720 images divided into training and testing datasets with 3626 and 2782 images, respectively. The TuSimple dataset is unique because it annotates complete lane boundaries instead of lane markings. As a result, this dataset is perfect for this research.

Moreover, this dataset is used as a training and testing dataset in [73], with about 3600 training images and 2700 testing images. The author stated that the TuSimple dataset comprises a variety of weather scenarios and is a massive dataset for measuring lane detection performance. Furthermore, this literature presented a strategy using the spatially convolutional neural network (SCNN) method [19]. Although the TuSimple dataset includes various road situations, including straight lines, curving lanes, splitting and merging lanes, and shadows, only straight and curvy lane scenarios were employed in this study.

This dataset was also utilized in the literature [36] to evaluate their strategy. Next, Z. M. Chng *et al*. run lane detection experiments on the TuSimple test sets in [56]. According to the literature, this dataset is relatively simple, taken during the daytime along highways in excellent or moderate weather, and contains ground truths annotated on the last frame of each twenty-frame clip. The author manually selects the lane markers demarcating the active lane for detection and comparison in the tests for each frame with ground truths labeled. The TuSimple lane dataset consists of 3,626 picture sequences. These are highway driving scenes from the driver's perspective. Each sequence contains 20 uninterrupted frames captured over the one-second time frame. The last frame of each series, i.e., the 20th image, is labeled with lane annotations. In addition, this literature adds labels to every 13th frame in each sequence to augment the dataset. Finally, [106] used the TuSimple lane dataset on the lane detection task to train and test deep learning-based techniques.

2) KITTI DATASET

The KITTI [134] benchmark is also popular data for road scenes. It contains various information regarding the road scene, including color pictures, stereo images, and laser point data. Jannik Fritsch and Tobias Kuehnl of Honda Research Institute Europe GmbH generated the KITTI Vision benchmark dataset [134]. There are 289 training and 290 test images in the road and lane estimate benchmark. Urban unmarked (UU), urban multiple marked (UMM), urban marked (UM), and hybridization of the three categories are the four categories that the pictures of road scenes fall into. The training dataset consists of 98 images; meanwhile, the testing dataset consists of 100 images. Ground truth was created in the KITTI dataset by manually annotating the images. It is offered for two types of road terrain: the road area (all lanes combined) and the lane (the current lane where the vehicle is traveling). For example, S. Shirke & R. Udayakumar [55] employed the KITTI dataset for region-based segmentation using an iterative seed approach for multilane identification. Aside from that, in another article, S. Shirke and R. Udayakumar [67] also used the KITTI vision benchmark dataset in their experimentation. Next, this public dataset KITTI also was then used to validate the algorithm's performance in [38]. Last but not least, P. Lu *et al*. [27] used the benchmark's testing dataset to validate the suggested technique.

3) CALTECH LANE DATASET

This dataset [135] contains four video clips captured throughout Pasadena, California, at distinct intervals of the day. The resolution of each video clip is 640 x 480 pixels and includes varying lighting and illumination situations, lane markings, sun glint, pavement types, shadows, crosswalks, and congested environments. In addition, this dataset also consists of urban streets, both straight and curved [102] was tested using the Caltech [135] dataset. Aside from that, the proposed methodology by B. Akbari *et al*. [19] was compared to two model-based methods using the Caltech Lane dataset. The author used about 1,224 labeled datasets in this literature, with 4,172 lanes extracted from four video clips collected from numerous urban roadways.

4) CITYSCAPES DATASET

Cityscapes' high-resolution and finely labeled training images [136] are well-known. On the other hand, this data offers semantic segmentation labels but not lane information [28]. Next, the author of [15] uses the Cityscapes dataset to test the network for broad semantic segmentation and multi-category lane line semantic segmentation tasks. The semantic comprehension of urban street sceneries from the perspective of a car is the focus of this literature. The collection contains 5000 photos with high-quality pixel-level annotations with 2975 training datasets, 500 validation datasets, and 1525 test datasets.

5) APOLLOSCAPE DATASET

Apollo has six separating markings, four guiding markings, two stopping lines, 12 turning markings, and other pixel-level lane markings and lane characteristics [28]. With about 19040 photos, this is a vast data collection (training sets are 12400, validation sets are 3320, and test sets are 3320, respectively). In addition, a stopping line, a zebra line, a single solid line, a single dash line, a double solid line, and other semantic segmentation information can also be found on the road. However, some ground area near lane lines is easily mistaken for lane markings [28]. The following works used the ApolloScape dataset for training and testing. For instance, in [15], the author analyses the network for generic semantic segmentation tasks and multi-category lane line semantic segmentation using the ApolloScape dataset. As the author knows, this dataset is challenging to work with because it includes high-quality pixel-level ground truth of over 110 000 frames and lane elements such as six separating markings, four guiding markings, two stopping lines, and 12 turning markings, among others. Furthermore, the author employs multi-class training in this experiment. ApolloScape offers three different datasets; however, they only used one for the lane detection task in this literature.

6) CULANE DATASET

The CULane dataset can be considered more challenging, and many datasets include normal conditions and eight complex settings such as crowded, night, and online. On the other hand, the TuSimple dataset is more straightforward than CULane. Therefore, several frames in CULane lack lane markers (e.g., at light traffic crossroads). The studies in [56] were carried out on test sets of one of the most widely used

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

and extensively utilized lane detection datasets [5]. The CULane train set is used to pre-train the models. Furthermore, this dataset comprises several challenging driving scenarios and ground truths annotated on all frames (e.g., congested city streets and night scenes with poor lighting). Besides, it is a simple dataset collected during the daytime along highways in excellent or moderate weather.

Most public datasets for lane detection, such as TuSimple, Caltech, Kitti, CULane, and Cityscapes, are currently proposed for urban roadways. The TuSimple is widely used in the literature, as evidenced by the publications chosen. It is the most often used dataset among academics in lane detecting studies. Tusimple has been used to test many algorithms [1], [5]– [20], [21], as it was the largest lane detection dataset before 2018. This dataset contains 3626 training photos and 2782 testing images on highway roads. It is intended for ego-road lane recognition; however, it does not distinguish between lane marker kinds or offers space between lanes. TuSimple, on the other hand, is a simple dataset collected during the daylight along highways in excellent or moderate weather, with ground facts only labeled on the last frame of each clip of twenty frames [18]. Caltech is the second most used dataset for lane detection. The Caltech Lanes dataset contains four video sequences (or sub-datasets) in urban settings, totaling 1225 images, which have been used in some previous research [6], [9], [13], [21]–[28]. Aside from that, the Kitti and CuLane datasets are well-known online datasets for lane detecting tasks. The Kitti road has two sorts of annotations: road segmentation, which covers all lanes, and ego-lane, which designates the lane in which the car is presently moving. For examples of past research that used this dataset, see [21], [28], and [33]. CULane, on the other hand, features various challenging driving circumstances, including congested roads or highways with low lighting. As a result, it is rarely preferred by researchers for detecting the lane. [1], [8], [11], [16], [18] are some of the algorithms that use this dataset. Some CULane frames lack lane markers (for example, crossing traffic light crossings) [18].

Furthermore, there are usually three sets of dataset partition for training the models: training set, validation set, and test set. The training set will be used to fine-tune the model's parameters. Meanwhile, the validation set (which can be ignored if just one model is supplied) and the test set (which will be used to quantify the model's accuracy) will be used to compare alternative models applied to that data. Normally, the proportions of these partitions are 70/20/10. The divisions of the dataset from multiple prior studies provided in this study were presented in this portion of the SLR, as illustrated in Table VII. The division of the dataset consists of a training set, validation set, and test set in percentages. The popular dataset, such as TuSimple, are mostly divided into 60% training and 40% testing set. Meanwhile, the Kitti dataset is divided into 50% training and 50% testing. Next, the NuScenes data set is divided into 90% training and 10% validation. Therefore, there is also a dataset used by previous researchers where the data distribution is not the same. For example, CULane dataset distributed to 60% training, 10% validation and 30% testing [37], 65% training, 10% validation and 25% testing [81], and 75% training and 25% testing set [56]. The previous CamVid, dataset has been divided into 80% training and 20% testing [66], 60% training and 40% testing [120].

D. LEARNING OUTCOMES FROM THE RQs

According to the literature analysis, it is shown that in just four years, the development of the lane detection task from the traditional-based method, which requires many pipeline processes, to the existence of the Artificial intelligence field, which is an intense learning-based strategy, the study will be easier and more efficient. For instance, deep learning algorithms have a high degree of generalization and learn essential aspects of the driving environment. However, there is always space for development in speed and accuracy, particularly in adverse weather situations, when applying the deep learning-based approach. Thus, several works of literature have advocated the integration of this method. Therefore, the integration of DL and attention mechanism become the state-of-the-art approach still new in this field as it just began to introduce in 2020. Therefore, only a few studies in the literature have studied lane detection using deep learning and the attention mechanism. The attention mechanism was previously utilized primarily in natural language processing (NLP), but it is now broadly used in computer vision, particularly in the medical field. Thus, it can be explored more in the automation field.

Next, the self-collected dataset can be acquired using various sensors, including cameras. It has been found that the camera is the most popular sensor for lane detection applications. Cameras have improved in reliability and are now likely to capture any situation on the road from any angle. In addition, vision sensors are becoming more effective and cost-efficient due to recent deep learning techniques. Moreover, due to the widespread use and efficiency of camera sensors, deep learning algorithms can learn the crucial features and characteristics of the driving environment across multiple layers in the model. Next, there are primary benefits of using the camera. This sensor delivers extensive information about the surroundings and is currently the most cost-effective and dependable method for automotive applications.

Besides the camera, LiDAR, radar, HD map, simulator, and OSM are also used as equipment for data collection. It is due to the camera sensor being affected by light conditions, which necessitates a filtering process. Therefore, during the data collection using the camera, the driver must ensure that the range distance between the experiment vehicle and the front car is always suitable. Then, the same range distance for better quality input image will be obtained. Other than that, there are dangerous and consumes time to collect the dataset

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

using a camera, especially during the rainy/monsoon season. Especially in Southeast Asia, there is a time when heavy drop rain will continue for the whole week. In addition, it is difficult to collect the dataset in an urban area at a specific time, for example, during peak hours, when there would be many vehicles on the road and stuck in traffic jams. However, cameras are less expensive than LiDAR. Meanwhile, OSM data is devoid of precision information.

Next, the simulator is commonly used for modeling lane detection and used as equipment for data collection. There are several advantages when using the simulator to collect the dataset for training, testing, and validation. One of the advantages is that it is not time-consuming and non-dangerous because it is not involved with the physical and natural environment. Therefore, it can create many conditions, especially extreme conditions such as rain, snow, fog, etc.

Other than the self-collect dataset, there are also several available online datasets in the market. Various repositories exist for a dataset on lane detection, such as the TuSimple dataset, KITTI vision benchmark dataset, CULane dataset, Cityscapes dataset, and Caltech dataset. This dataset is straightforward, has a variety of image situations, and has already been labeled for the training dataset. TuSimple is the most popular dataset since it incorporates different road conditions, including straight lines, curving lanes, splitting and merging lanes, and shadows. Not only have that, but the TuSimple dataset also includes lane detecting images with lower illumination.

Furthermore, the TuSimple dataset collects data from roads in fair or moderate weather, with two lanes/three lanes/or more lanes, and a variety of traffic scenarios, including clear lane lines with excellent image quality, no blur, and relatively simple identification challenges. Unfortunately, even though several ready companies with Level 5 autonomous cars are claimed, the available data for extreme conditions is still limited. The learning results from RQs 1, 2, and 3 are summarised in Table VI. The table contains the technique deployed for 102 selected publications with the dataset type and equipment for the self-collect dataset.

E. GENERAL DISCUSSION ON ADDRESING THE SPECIFIC ISSUES BASED ON COMPUTER VISION TECHNIQUE

Most geometric modeling/conventional approaches rely on or follow pre-processing feature extraction, lane model fitting, and lane tracking to detect the lane. For lane detection tasks, image pre-processing is required to determine the quality of features. In addition, this approach needs manually alter the parameters, although this procedure is efficient and uncomplicated. Furthermore, previous methods based on handcrafted features to detect lanes are limited in scenarios using edge, texture, or color information, which requires complicated post-processing modules to perform. Likewise, in many complex procedures, these approaches function inadequately. Therefore, the traditional computer vision (CV) techniques are time-consuming and resource-intensive and rely on complicated algorithms to analyze the delicate aspects of lane images. In addition, the number of lanes is frequently not fixed, and techniques for detecting lanes are sometimes erroneous. Straight lines, for example, are treated as lanes by Hough transform-based algorithms, which may cause streetlamps to be mistaken for lanes. Furthermore, poor weather, such as rain, will impact lane detecting. Likewise, inadequate lighting and a night setting will produce poor results. However, there are yet no practical solutions for dealing with such issues. As a result, conventional approaches are ineffective in detecting lanes in complex traffic situations. In addition, it must work in real-time. Most algorithms, however, suffer from a lack of this purpose. Therefore, traditional techniques have yielded significant results. However, they have several limitations: (1) lane detection is challenged in varying weather conditions and illumination. Furthermore, previous methods lack a consistent framework for detecting various scenes and (2) the inefficiency of using images due to label noise.

Following that, due to the advancement of deep learning, numerous solutions have been suggested to enhance the achievement of computer vision works in contrast to conventional approaches. Despite the prevalence of camera sensors, deep learning algorithms offer a high degree of generalization and learn the essential elements of the driving environment across multiple layers. In contemporary state-of-the-art lane detection techniques, convolutional neural networks (CNNs) are also used to develop deep learning models. CNN is also created for image classification problems in deep learning-based technology, in which it can extract features from the images it receives. However, the image's output is one-dimensional data that can only forecast which images belong to which sorts of objects. Furthermore, numerous low-level characteristics were lost in the pooling layers of CNN. As a result, input changes might cause convolution neural networks to lose robustness. Scale, viewpoint, and backdrop clutter are examples of these variances. Furthermore, while these models perform admirably on train and test inputs, they perform poorly on unknown datasets from various contexts. The FCN network can overcome these issues and detect more accurate two-dimensional data. Even the deep learning-based technique offers numerous advantages. However, they have a high computing cost, which can sometimes increase training loss and result in a vanishing gradient issue [38].

In the past, advanced detecting algorithms such as deep learning have outperformed traditional methods in complex scenarios, but they have limitations. For example, despite the importance of multilane detection, only a limited number of lanes can be detected, and the cost of detection time is frequently prohibitive. Therefore, various factors influence lane detection tasks, including specific complex traffic scenarios.

Attention mechanisms have improved NLP and CV extensively. The employment of an attention mechanism improves feature localization in the feature map and eliminates the need for post-processing. Therefore, as lanes are long and thin for lane detection, there are considerably fewer annotated lane pixels than background pixels, which is challenging for a model to learn. Hence, the attention processes in feature maps can emphasize crucial spatial information. The attention mechanism, in particular, can boost the weighted information of lane line objectives while reducing unnecessary data. It adds to the complexity of network learning. However, as the author is aware, little research has been done on using the attention mechanism in lane detecting tasks. In this research area, many different forms of attention mechanisms can be used at the same time. As a result, the study's future direction can be investigated by applying another type of attention mechanism that has yet to be deployed.

F. ISSUE ON TECHNIQUE RELATED TO DATA

The existing ADAS act as a driver's aid and many issues still need to be addressed or improved to achieve the objective of safe and enjoyable autonomous driving on real-world roads. In a real-world scenario, a lane recognition system should continue to work throughout the year, regardless of whether it is sunny or cloudy, day or night, summer or winter, urban or rural, crowded or clear, and so on. The main challenge is to make the lane recognition approach resilient and prosperous under various driving conditions.

From the literature's selection, there are several issues on lane detection techniques that are related to data, such as:

a) Extremely **imbalanced data set problem** because the backdrop class contains the majority of the lane pixels in the image. In addition, the amount of backdrop pixels is significantly more than the number of lane pixels due to the lane's slenderness. It may be challenging to pick up on such characteristics [1]. Aside from unbalanced data, the quality of acquired data and annotations also restricts the capacity of various methods [2]. As a result of the limitations imposed by available datasets, lane approaches developed on one dataset are unlikely to be applied to another. To address this issue, state-of-the-art transfer learning and attention mechanisms must be implemented. Aside from that, a more generic dataset that replicates real-world road conditions should be investigated for the confined dataset. Furthermore, as this sector develops, more data sets are projected to become available for researchers, particularly with the advent of entirely autonomous cars [3]. However, researchers are also hindered by the lack of datasets, necessitating the creation of new databases to allow for additional algorithm testing. The new databases can be created using synthetic sensor data from a test vehicle or by generating driving scenarios using a commercially available driving simulator app. Similarly, more research is needed in the following areas.

b) **Variation and changeable lane markings**. With the vast diversity of lane markers, the complex and changing road circumstances, and the lane markings' inherent thin properties, Some scenarios, such as no line, shadow occlusion, and harsh lighting conditions [1], provide few or no visual signals. Therefore, detecting the lanes from the image in these scenarios can be difficult. According to the findings, traditional approaches work in a controlled environment and have numerous problems regarding robustness difficulties caused by road scene fluctuations. In addition, the lanes' inconsistency, curvature, and varied lane patterns make detection much more difficult. Daytime has gotten a lot of attention in the past, but nighttime and rainy situations have gotten less attention. Furthermore, it is apparent from the literature that, in terms of speed flow conditions, they have previously been examined at speeds ranging from 4 km/h to 80 km/h, with high speed (above 80 km/h) receiving less attention. Occluding overtaking vehicles or other objects and excessive illumination make lane identification and tracking difficult. Although reflector lanes are specified with several colors, lane markings are usually yellow and white. The number and width of lanes vary per country. There may be issues with vision clarity due to the presence of shadows. The visibility of the lane lines was reduced due to various weather conditions such as rain, fog, and snow. Visibility may be decreased in the evening. The performance of lane detection and tracking algorithms suffers due to these issues in lane recognition and tracking. As a result, developing a dependable lane detection system is a difficult task.

c) **Interference and illumination variations.** According to [4], lane-like interferences, such as guardrails, railways, utility poles, pedestrian sidewalks, buildings, and so on, will interfere with the existing traditional method, such as the HT-based algorithm. As a result, it has struggled in various challenging settings, including lane kinds, road surfaces, nighttime, and other environmental factors (shadow, rain, etc.). When a vehicle drives at night, the intensity of the region illuminated by headlights, for example, is several orders of magnitudes higher than the backdrop. As a result, even though the lane markers contrast nicely with the road surface in human vision, portions of lanes are overexposed. The host car then casts its shadows on the road surface as it enters or exits a tunnel or drives beneath a bridge. As a result, the road may have complicated painted road surface markings, utility lines, and buildings, which can cause the HT-based lane recognition algorithm to provide misleading edges and textures. On rainy days, reflection from the wet road may induce glare and image overexposure, resulting in lane detection failure in some instances. In addition to lane-like interferences, lighting fluctuations make dividing line recognition more challenging. Under artificial light, the system failed to recognize road lane characteristics in bright or wet road conditions with significant reflection on rainy days. Using assumptions to delete the misleading edges far from the host lane during the picture pre-processing step may be one technique to lower the false-positive rate under such scenarios. Another option is to employ feature-based machine learning algorithms. This could be one of the areas where future research could be conducted. However, such approaches would struggle on roads not included in the training set, and they would tend to overfit to sounds in pictures and lane markers. As a result, classic techniques such as model predictive controller (mpc) have worse performance in bad weather and pose issues in controlling high illumination or shadows, according to [3].

G. UNCERTAINTIES MANAGEMENT

Working with inaccurate or incomplete information is what uncertainty entails. This study contains numerous sources of uncertainty, including data noise and an imprecise model. The solution systematically evaluates multiple keys until an excellent or good-enough set of features and methods is found for a given problem.

a) **Noise**. Noise is the term for variation in an observation. Both the inputs and the outputs are affected by this unpredictability. Genuine data, like the real world, is a tangled mess. Therefore, maintaining skepticism about data and developing techniques to anticipate and battle uncertainty is crucial. The solution to this problem is to invest some time analyzing data statistics and creating visualizations to aid in identifying those anomalous or unusual cases: this is what data cleansing is all about.

b) **Incomplete coverage of the domain**. A random sample is a set of observations picked randomly from a domain with no systematic bias. A certain amount of bias will always exist. This arises when a model lacks sufficient data and knowledge, commonly occurring when there aren't enough samples to train the Artificial Intelligence. While some bias is inherent, uncertainty grows if the sample's degree of variance and bias is an unsatisfactory representation of the task for which the model will be utilized. For example, in lane detection, researchers may detect a lane in a highway area only if the road is in good condition and there are few vehicles present except during rush hour. Aside from that, lane detection in normal situations is far easier than in extreme conditions. The painted lane marking is chosen at random. However, it can only be used in one instance. The scope can be expanded to include highways, cities, rural areas, and normal, rainy, and foggy circumstances. The sample must have an acceptable amount of variance and bias to represent the task for which the data or model will be utilized. There will never be all of the observations in any of the initial investigations. This implies that some cases will always go unnoticed. There will be areas of the problem domain that are not covered. Two options are splitting the dataset into train and test sets or using resampling methods like k-fold cross-validation. This technique can be used to deal with ambiguity in the dataset's representativeness and to assess the performance of a modeling procedure on data that isn't included.

H. ANALYSIS OF PERFORMANCE EVALUATION METRICS & ITS SIGNIFICANCE

Various performance indicators are available, but the most frequent are accuracy, precision, F-score, and receiver operating characteristic (ROC) curves. If the dataset is balanced, the accuracy rate should reflect the algorithm's global output. The accuracy demonstrates the accuracy of optimistic predictions. The lesser the amount of "false alarms," the higher the accuracy. The recall, also known as the true positive rate (TPR), is the proportion of positive cases that the algorithm accurately detects. As a result, the better the recall, the more accurate the algorithm finds positive instances. The F1 score is the harmonic mean of Precision and Recall, and because they are merged into a single metric, it may be used to compare algorithms. The harmonic mean is employed instead of arithmetic since it is more sensitive to low values. As a result, if an algorithm is accurate and has a high recall, it has a decent F1 score. These parameters can be calculated as individual metrics for each class or overall metrics for the algorithm.

I. CROSS-VALIDATION FOR EVALUATING AND COMPARING MODULES

Cross-validation is a technique for testing how well a statistical analysis applies to a different dataset. Typically, the model is trained on a known dataset. This dataset is referred to as the training dataset. However, the model must work on an unknown dataset in real-time. Cross-validation is used to see how well a prediction model works with an anonymous dataset. The model may appear to have a high degree of accuracy when the original validation division does not reflect the entire population. However, it will be of little help in practice because it can only work with limited data collection. When it comes across data outside of its scope, the system cannot recognize it, resulting in poor accuracy. It is verified how accurate the model is on many diverse subsets of data when cross-validation is employed in machine learning. As a result, it ensures that it generalizes well to data collected in the future. It enhances the model's accuracy. Cross-validation might help you avoid overfitting and underfitting. When a model is trained "too well," overfitting develops. It occurs when the model is sophisticated and has a large number of variables in comparison to the number of data. In such cases, the model will perform admirably in training mode but may not be accurate when applied to new data. It is because it is not a generalized model. Underfitting happens when the model does not fit the training data instead of overfitting. As a result, it is unable to generalize to new data. It's because the model is simple and lacks sufficient independent variables. In data analysis, both overfitting and underfitting are undesirable. It should always strive for a balanced approach or a 'just right paradigm. Overfitting and underfitting can both be avoided by cross-validation. Machine learning necessitates extensive data analysis. Cross-validation is a great way to get the machine ready for real-world circumstances. As a result, the system is prepared to take in new data and generalize it to make correct predictions. However, to the authors' knowledge, previous research in the lane detection sector does not generally discuss or describe any cross-validation for evaluation. It is possible to state that it is a biased experiment that requires additional examination in this sector.

J. LIMITATION OF SYSTEMATIC LITERATURE REVIEW BASED ON RESEARCH QUESTIONS

Referring to the research questions, RQ1, RQ2 and RQ3, there are existing of certain limitations as listed as follows: -

(a) RQ1 – The results from previous research mostly demonstrate that in most circumstances, lane detection accuracy is about 96 percent under normal conditions. Heavy rain, on the other hand, significantly impacts the efficiency of lane marker detection. In addition, external factors such as weather, visual quality, shadows, and blazing, as well as internal factors such as lane marking that is too narrow, too broad, or unclear, degrade the performance. Moreover, it has been observed that the system's performance suffers due to unclear and deteriorated lane markers. Therefore, one of the most significant issues with current ADAS is the ambient and meteorological environments substantially impacting the system's functionality.

(b) RQ2 - Regarding lane marking, camera quality is crucial, and an adjacent vehicle may obscure the lane signs during overtaking. Therefore, the algorithm's accuracy is determined by the camera used. Images were captured using monocular, stereo, and infrared cameras. From the literature, a stereo camera outperforms a monocular camera. RQ3 – Approximately 60% of the researchers have used self-collected datasets in their research.

K. LIMITATION, FUTURE SCOPE AND CONTRIBUTIONS OF THE CURRENT WORK

The limitations and future scope of the current work can be categorized into **methods**, **dataset**s, and **model network architecture**.

1) Methods

Limitations: Most geometric modeling/conventional approaches rely on or follow pre-processing feature extraction, lane model fitting, and lane tracking to detect the lane. For lane detection tasks, image pre-processing is required to determine the quality of features. In addition, this approach needs manually alter the parameters, although this procedure is efficient and uncomplicated. Furthermore, previous methods based on handcrafted features to detect lanes are limited in scenarios using edge, texture, or color information, which requires complicated post-processing modules to perform. Likewise, in many complex scenarios, these approaches function inadequately. Therefore, the traditional computer vision (CV) techniques are time-consuming and resource-intensive and rely on complicated algorithms to analyze the delicate aspects of lane images. In addition, the number of lanes is frequently not fixed, and techniques for detecting lanes are sometimes erroneous. Straight lines, for example, are treated as lanes by Hough transform-based algorithms, which may cause street lamps to be mistaken for lanes.

Furthermore, poor weather, such as rain, will impact lane detecting. Likewise, inadequate lighting and a night setting will produce poor results. However, there are yet no practical solutions for dealing with such issues. As a result, conventional approaches are ineffective in detecting lanes in complex traffic situations. In addition, it must work in real-time. Most algorithms, however, suffer from a lack of this purpose. Therefore, traditional techniques have yielded significant results. However, they have several limitations: (1) lane detection is challenged in varying weather conditions and illumination.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

TABLE VI
The Learning Results From RQs 1, 2, and 3

Number	References	Technique	Dataset	Equipment For Self-Collect Dataset
1	[97]	Geometric modelling	Self-collect	Camera
2	[98]	Geometric modelling	Self-collect & Caltech dataset	Camera
3	[99]	Geometric modelling	Self-collect	Camera
4	[100]	Geometric modelling	Self-collect	Monocular camera
5	[101]	Geometric modelling	Self-collect	Camera
6	[102]	Deep learning (CNN) + geometric modelling	Self-collect & Caltech dataset	Camera
7	[103]	Geometric modelling	Self-collect	Camera
8	[104]	Geometric modelling	Self-collect	Camera
9	[105]	Geometric modelling	Self-collect	Camera
10	[106]	Deep learning (semantic segmentation based CNN) + geometric modelling	Self-collect & TuSimple dataset	Camera
11	[107]	Geometric modelling	Caltech dataset	Camera
12	[108]	Two Serial Deep Learning (Haar cascades + Faster R-CNN)	Self-collect	Collecting training dataset on Unity3D simulator
13	[109]	Deep learning (CNN) + Geometric modelling	Road Vehicle Dataset (RVD) & Caltech dataset & TuSimple dataset	-
14	[22]	Geometric modelling	Self-collect	Camera
15	[110]	Geometric modelling	Self-collect	Camera
16	[111]	Geometric modelling	Self-collect -Large Variability Road Images database (LVRI)	Camera
17	[72]	Deep learning (Resnet)	TuSimple dataset	-
18	[73]	Two Serial Deep Learning (FR-IQA gradient-guided deep networks + RNN)	TuSimple dataset	-
19	[74]	Deep learning (cnn) + geometric modelling	Self-collect	ov7725 camera
20	[75]	Geometric modelling	Self-collect	Camera
21	[76]	Deep learning (cnn) + geometric modelling	Self-collect	Camera
22	[77]	geometric modelling	Self-collect & Caltech dataset	Camera
23	[78]	Deep learning (LaneNet)	Self-collect	Camera
24	[79]	Geometric modelling	KITTI dataset	Camera
25	[80]	Deep learning (semantic segmentation based U-net (FCN)) + geometric modelling	Self-collect	Collecting training dataset on CARLA simulator
26	[81]	Deep learning (semantic segmentation based CNN)	Self-collect- CVPR 2017 TuSimple dataset & CULane dataset	-
27	[82]	Geometric modelling	Self-collect	Camera
28	[83]	Geometric modelling	Self-collect	Camera
29	[84]	Geometric modelling	Self-collect	Low-cost image sensor (dashcam)
30	[85]	Geometric modelling	Self-collect	Camera
31	[86]	Deep learning (image cascade network ICNet)	Cityscapes dataset	-
32	[87]	Geometric modelling	The California Institute of Technology has published a road dataset, as well as one collected by Beijing Union University (BUUD).	-
33	[88]	Geometric modelling	Caltech Dataset	-
34	[89]	Deep learning (CNN) + geometric modelling	-	-

35	[90]	Deep learning (SegNet) + Machine learning (Bayesian learning)	Self-collect	A high resolution automotive radar prototype is used for data collection
36	[91]	Geometric modelling	Self-collect	Camera
37	[92]	Geometric modelling	Self-collect	Camera
38	[93]	Deep learning (semantic segmentation based CNN) + geometric modelling	-	-
39	[94]	Geometric modelling	Self-collect	Camera
40	[95]	Two Serial Deep Learning (Haar cascades + Faster R-CNN)	Self-collect	Camera
41	[96]	Machine learning (Bayesian learning) + geometric modelling	Self-collect	Camera
42	[45]	Geometric modelling	Self-collect	LiDAR data and OpenStreetMap (OSM) data
43	[46]	Machine learning (Bayesian learning) + Geometric modelling	ROMA dataset	-
44	[47]	Deep learning (YOLO v3)	Self-collect	Over 25,000 pictures of various road surface markers were acquired in mass from Google Images.
45	[25]	Geometric modelling	Self-collect	Camera
46	[48]	Geometric modelling	Self-collect	Camera
47	[49]	Geometric modelling	Self-collect	Vision or camera data
48	[50]	Deep learning (DarkSCNN) + Geometric modelling	Self-collect	A camera and a digital map
49	[51]	Deep learning (CNN) + Machine learning (SVM)	Self-collect & TuSimple dataset	-
50	[52]	geometric modelling	Self-collect	Camera
51	[137]	Geometric modelling	Self-collect	-
52	[54]	Geometric modelling	Self-collect	Around Lafayette, Indiana, authors recorded the images of Interstate Highway 65 and also local roads.
53	[55]	Geometric modelling	KITTI dataset	-
54	[56]	Two Serial Deep Learning (SCNN & RONELD (robust nn result improvising for vigorous lane detection)	TuSimple dataset & CULane dataset	-
55	[57]	Deep learning (R-CNN)	Self-collect	Capture live video and process it to extract 3D information
56	[58]	Geometric modelling	Self-collect	Camera
57	[59]	Two Serial Deep Learning (CNN, ERFNet & H-Net)	TuSimple dataset	-
58	[60]	Geometric modelling	Public dataset	-
59	[61]	Geometric modelling	Self-collect	Monocular camera lens
60	[62]	Geometric modelling	Self-collect	Monocular camera lens
61	[63]	Two Serial Deep Learning (CNN + RCNN + ConvLSTM)	Self-collect & TuSimple dataset	A moving car was used to create this photographic sequence. A colour camera is positioned along the centre line of the front-view mirror inside the vehicle.
62	[64]	Geometric modelling	Self-collect	Camera
63	[65]	Geometric modelling + Machine learning (Extreme learning machine (ELM))	-	-
64	[66]	Deep learning (LiteSeg + MobileNetV2) + geometric modelling	Self-collect & CamVid dataset	Using a Kinect camera integrated in a 1:7 RC car, collect data in a limited driving scenario.
65	[67]	Deep learning (DCNN) + geometric modelling	KITTI dataset	-

66	[68]	Geometric modelling	-	-
67	[69]	Deep learning (CNN)	On the Berkley Deep Drive dataset	-
68	[70]	Two Serial Deep Learning (encoder-decoder, ResNet + DenseNet)	Self-collect	A dataset including 314 orthoframe images of Estonian highways, each with a resolution of 4096 by 4096 pixels.
69	[71]	Geometric modelling	-	-
70	[15]	Deep learning (semantic segmentation-based encoder-decoder) + attention mechanism	Self-collect & ApolloScape dataset & Cityscapes dataset	-
71	[16]	Geometric modelling	Self-collect	The WiFi sports camera sensor
72	[17]	Geometric modelling	-	-
73	[18]	Geometric modelling	-	-
74	[19]	Geometric modelling	Self-collect & Caltech dataset & TuSimple dataset	Camera
75	[20]	Deep learning (semantic segmentation based Deeplabv3plus) + geometric modelling	Self-collect	Camera
76	[21]	Deep learning (encoder-decoder) Encoder- VGG-16, MobileNet, and ShuffleNet Decoder-Unet	Self-collect dataset & TuSimple dataset	Camera
77	[22]	Geometric modelling	-	-
78	[10]	Deep learning (instance segmentation based LaneNet) + attention mechanism	TuSimple dataset and CULane dataset	-
79	[11]	Deep learning (encoder-decoder) + attention	Det dataset	-
80	[23]	Deep learning (FCN-UNet & SegNet) + geometric modelling	nuScenes dataset	-
81	[24]	Deep learning (CNN) + geometric modelling	TuSimple dataset and CULane dataset	-
82	[25]	Deep learning (Convoluted NNs utilizing Spatial Transformer Networks) + geometric modelling	German Traffic Sign dataset	-
83	[26]	Machine learning (functional link artificial neural network (FLANN)+ geometric modelling	Caltech dataset & KITTI dataset & ROMA dataset	-
84	[27]	Machine learning + geometric modelling	Self-collect	The input footage utilised in this investigation was shot on normal North American highways.
85	[28]	Geometric modelling	The suggested system would be trained using dataset from the Cityscapes, Vistas, and Apollo datasets, and then its efficiency would be evaluated. Using TuSimple dataset, Caltech dataset, KITTI dataset, and X-3000 dataset	-
86	[29]	Deep learning (convLSTM)	Self-collect	Camera
87	[30]	Geometric modelling	-	-
88	[31]	Deep learning (LaneFCNet) + geometric modelling	Self-collect & Cityscapes dataset	Camera
89	[12]	Geometric modelling	Self-collect & Caltech dataset	Camera
90	[32]	Deep learning (encoder-decoder) + attention mechanism	TuSimple dataset and Caltech dataset	-
91	[33]	Geometric modelling	-	-
92	[34]	Geometric modelling	KITTI dataset	-
93	[35]	Geometric modelling	Self-collect	Camera
94	[36]	Deep learning (multi-stage Convolutional Neural Network (CNN))	TuSimple dataset	-
95	[37]	Geometric modelling	TuSimple dataset & CULane dataset	-

96	[38]	Deep learning (Semantic segmentation based encoder-decoder) + UNet + Resnet-50))	KITTI dataset	-
97	[39]	Geometric modelling	Self-collect	Stereo camera
98	[40]	Geometric modelling	Self-collect	LiDAR point cloud
99	[138]	Two Serial Deep Learning (Spatial CNN (SCNN) + multi spatial convolution block (MSCB)	PSV dataset & TSD dataset	-
100	[139]	Deep learning (encoder-based Generative Adversarial Network (eGAN) + Geometric modelling	-	-
101	[43]	Deep learning (LLNET (CNN))	TuSimple dataset	-
102	[44]	Deep learning (FCN)	-	-

TABLE VII
Dataset Partition

References	Type of Dataset	Training set (%)	Validation set (%)	Test set (%)
[11]	DET	50	16	33
[16]	ApolloScapes	70	15	15
[23]	NuScenes	90	10	-
[24]	CULane, TuSimple, Beijing	70	20	10
[31]	German Traffic Sign	70	20	10
[31]	TuSimple	60	-	40
[31]	Kitti	50	-	50
[31]	CityScapes	85	-	15
[36]	TuSimple	60	-	40
[37]	TuSimple	60	-	40
[37]	CULane	60	10	30
[38]	Kitti	50	-	50
[23]	NuScenes	90	10	-
[47]	Self-collect	90	-	10
[140]	TuSimple	60	-	40
[55]	Kitti	50	-	50
[59]	TuSimple	60	-	40
[66]	CamVid	80	-	20
[67]	Kitti	50	-	50
[72]	TuSimple	50	5	45
[73]	TuSimple	60	-	40
[81]	CULane	65	10	25
[90]	Self-collect	64	16	20
[56]	TuSimple	60	-	40
[56]	CULane	75	-	25
[120]	CamVid	60	-	40

Furthermore, previous methods lack a consistent framework for detecting various scenes and (2) the inefficiency of using images due to label noise. Following that, due to the advancement of deep learning, numerous solutions have been suggested to enhance the achievement of computer vision works in contrast to conventional approaches. Despite the prevalence of camera sensors, deep learning algorithms offer a high degree of generalization and learn the essential elements of the driving environment across multiple layers. In the past, advanced detecting algorithms such as deep learning have outperformed traditional methods in complex scenarios, but they have limitations. For example, despite the importance of multilane detection, only a limited number of lanes can be detected, and the cost of detection time is frequently prohibitive. Therefore, various factors influence lane detection tasks, including specific complex traffic scenarios. Attention mechanisms have improved NLP and CV extensively. The employment of an attention mechanism improves feature localization in the feature map and eliminates the need for post-processing. Therefore, as lanes are long and thin for lane detection, there are considerably fewer annotated lane pixels than background pixels, which is challenging for a model to learn. Hence, the attention processes in feature maps can emphasize crucial spatial information. The attention mechanism, in particular, can boost the weighted information of lane line objectives while reducing unnecessary data. It adds to the complexity of network learning. However, as the author is aware, only a little research has been done on using the attention mechanism in lane detecting tasks.

Future scope: In this research area, many different forms of attention mechanisms can be used at the same time. As a result, the study's future direction can be investigated by applying another type of attention mechanism that has yet to be deployed.

2) Dataset

Limitations 1: Extremely imbalanced data set problem because the backdrop class contains most of the lane pixels in the image. The amount of backdrop pixels is significantly more than the number of lane pixels due to the lane's slenderness. It may be challenging to pick up on such characteristics. Aside from unbalanced data, the quality of acquired data and annotations also restricts the capacity of various methods [2].

Future Scope 1: State-of-the-art mechanisms such as transfer learning and attention mechanisms can be implemented. Aside from that, a more generic dataset that replicates real-world road conditions can be investigated for the confined dataset. Furthermore, the new databases can be created using synthetic sensor data from a test vehicle or by generating driving scenarios using a commercially available driving simulator.

Limitations 2: Changeable Lane markings and illumination variations. The wide diversity of lane markers, the complex and changing road circumstances such as no line, shadow occlusion, provide few or no visible lane lines, the inconsistency of the lanes, the curvature of the lane, and the varied lane pattern makes detection much more difficult. According to the findings, traditional approaches work in a controlled environment and have numerous problems when it comes to robustness difficulties caused by road scene fluctuations. Furthermore, occlusion from overtaking vehicles or other objects and excessive illumination make lane identification and tracking difficult. Other than that, the visibility of the lane lines was reduced due to weather conditions such as rain, fog, and snow. The performance of lane detection and tracking algorithms suffers due to these issues in lane recognition and tracking. In addition, according to [4], lane-like interferences, such as guardrails, railways, utility poles, pedestrian sidewalks, buildings, and so on, will interfere with the existing traditional method, such as the HT-based algorithm. As a result, it has struggled in various hardy environments, including nighttime and other environmental factors (shadow, rain, etc.).

Furthermore, the host car then casts its shadows on the road surface as it enters or exits a tunnel or drives beneath a bridge. As a result, the road may have complicated painted road surface markings, utility lines, and buildings, which can cause the HT-based lane detection algorithm to provide misleading edges and textures. On rainy days, reflection from the wet road may induce glare and image overexposure, resulting in lane detection failure in some instances. In addition to lane-like interferences, lighting fluctuations make dividing line detection more challenging. Under artificial light, the system failed to recognize road lane characteristics in bright or wet road conditions with significant reflection on rainy days.

Future Scope 2: Employ feature-based learning models to control lousy weather and illumination and shadows issues.

3) Models Network Architecture

Limitations: Working with inaccurate and incomplete information is what uncertainty entails. This study contains numerous sources of uncertainty, including data noise and an imprecise model. Noise is the term for variation in an observation. Both the inputs and the outputs are affected by this unpredictability.

Genuine data, like the real world, is a tangled mess. Besides that, a random sample is a set of observations picked randomly from a domain with no systematic bias.

Nevertheless, a certain amount of bias will always exist. This arises when a model lacks sufficient data and knowledge, commonly occurring when there aren't enough samples to train the model. While some bias is inherent, uncertainty grows if the sample's degree of variance and preference is an unsatisfactory representation of the task for which the model will be utilized.

For example, in lane detection, researchers may detect a lane in a highway area only if the road is in good condition and there are few vehicles present except during peak hours. Aside from that, lane detection in normal situations is far easier than in extreme conditions. The painted lane marking is chosen at random. However, it can only be used in one instance. The scope can be expanded to include highways, cities, rural areas, and normal, rainy, and foggy circumstances. The sample must have an acceptable amount of variance and bias to represent the task for which the data or model will be utilized. There will never be all of the observations in any of the initial investigations. This implies that some cases will always go unnoticed. There will be areas of the problem domain that are not covered.

Future Scope: Invest some time analyzing data statistics and creating visualizations to aid in identifying those anomalous or unusual cases. This is what data cleansing is all about. Therefore, splitting the dataset into train and test sets or using resampling methods like k-fold cross-validation. This technique can be used to deal with ambiguity in the dataset's representativeness and to assess the performance of a modeling procedure on data that isn't included in the training.

Contributions: Combining deep learning approaches with other techniques yields significant performances. The merging of networks and attention mechanism was proposed to learn more discriminative features of road lanes than the stand-alone deep learning approach to significantly increase the detection accuracy of the road lane. These methods/innovations regarding more precise lane detection are necessary to enable a real-time lane detection system. Therefore, the model's accuracy and speed should be improved in normal and extreme conditions.

L. COMPARISON WITH ALREADY DONE REVIEW ARTICLES

This SLR is compared with the other review articles that have previously been completed. As a result of the SLR, it was discovered that most of the currently published research falls into one of the categories presented and discussed in Table VIII.

V. CONCLUSIONS AND FUTURE RECOMMENDATIONS

This review article is concluded by analyzing the outcomes and making recommendations for subsequent initiatives. This section describes all lane detection methods, self-collect dataset preparation equipment, the top three most popular online datasets, fundamental problems in this field, and the state-of-the-art that can be investigated for future research.

A. CONCLUSIONS

The analysis from this SLR shows that the selected literature used various methods and structures, with the input dataset being one of two types: self-collected or acquired from an online public dataset. In the meantime, the methodologies include geometric modeling and traditional methods, while AI includes deep learning and machine learning. CNN, FCN, and RNN are examples of deep networks and architectures. The use of deep learning has been increasingly researched throughout the last four years. Some studies used stand-alone deep learning implementations for a single lane detection problem or multiple lane implementations. Other than that, some research focuses on merging deep learning with other machine learning techniques and classical methodologies to improve efficiency.

On the other hand, recent advancements imply that attention-mechanism has become a popular strategy to combine with deep learning methods to increase performance. Using deep algorithms in conjunction with other techniques also showed promising outcomes. This SLR, we believe, will pave the path for more study to be accomplished to build more effective lane detection methods. In addition, more precise methods for use in real-world industrial settings are required. We plan to build on the findings of this study in the future, emphasizing creating a network with high-speed performance and efficiency that can be implemented in real-time.

B. FUTURE DIRECTIONS AND RECOMMENDATIONS

The following directions for future contributions to the discipline should be focused based on the findings of this SLR:

1. For exact feature learning, accurately labeled lane data is required for deep network training.
2. Increases the number of publicly available online public datasets that cover a wide range of scenes.
3. More imbalance management approaches, such as computational cost, speed performance, and algorithm/network training error should be investigated.
4. Combining deep learning approaches with other techniques yields significant results, which merits further investigation.

TABLE VIII
Comparison of This SLR with the Already Done Review Articles

Done Review Articles	SLR
1) According to the results of the study, the existing approaches provide good accuracy for high-quality images, but can produce unsatisfactory results under adverse environmental conditions such as fog, haze, noise, dust, etc. 2) The majority of existing approaches are optimal for straight lanes but are ineffective for curved roadways. 3) Since the majority of lane detection approaches are based on the basic Hough Transform, it can be improved to increase accuracy.	1) There are presently limited available data on extreme weather situations. Researchers are also hampered by a dearth of datasets, necessitating the production of new databases to enable additional algorithm testing. Using synthetic sensor data from a test vehicle or by producing driving scenarios with a commercially accessible driving simulator app, the new databases can be built. Additionally, additional research is required in the following areas: 2) Little study has been conducted on the application of the attention mechanism in lane detection tasks. In this area of study, numerous attention mechanisms can be employed. 3) Problems with vision clarity owing to the presence of shadows such as building and tree shadows, extreme weather conditions such as rain, fog, and snow, and complex or changing road conditions such as the absence of a lane and a complex lighting environment. As a result of these challenges with lane recognition and tracking, the performance of lane detection and tracking algorithms falls. Consequently, designing a reliable lane detection system is a challenging endeavor.

5. The merging of networks and attention mechanisms has improved performance, but additional research is needed.
6. They are developing approaches and technologies for lane detecting that are more efficient in speed and precision. The model's accuracy and rate under normal and extreme situations should be enhanced to enable real-time detection.
7. The computational load is reduced. Therefore, training time, memory, and CPU resources should all be minimized via efficient learning algorithms.

REFERENCES

[1] C. Y. Chan, “Trends in crash detection and occupant restraint technology,” *Proc. IEEE*, vol. 95, no. 2, pp. 388–396, 2007, doi: 10.1109/JPROC.2006.888391.

[2] Zehang Sun, G. Bebis, and R. Miller, “On-road vehicle detection: a review,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 694–711, 2006, doi: 10.1109/TPAMI.2006.104.

[3] K. A. Brookhuis, D. de Waard, and W. H. Janssen, “Behavioural impacts of advanced driver assistance systems–an overview,” *Eur. J. Transp. Infrastruct. Res.*, vol. 1, no. 3, pp. 245–253, 2001, doi: 10.18757/ejtir.2001.1.3.3667.

[4] K. Grove *et al.*, “Commercial Motor Vehicle Driver Performance with Adaptive Cruise Control in Adverse Weather,” *Procedia Manuf.*, vol. 3, no. Ahfe, pp. 2777–2783, 2015, doi: 10.1016/j.promfg.2015.07.717.

[5] U. Zakir *et al.*, “Current Collision Mitigation Technologies for Advanced Driver Assistance Systems – A Survey,” *PERINTIS eJournal*, vol. 6, no. 2, pp. 78–90, 2016, doi: 10.1105/tpc.15.01050.

[6] A. H. Eichelberger and A. T. McCartt, “Toyota drivers’ experiences with Dynamic Radar Cruise Control, Pre-Collision System, and Lane-Keeping Assist,” *J. Safety Res.*, vol. 56, pp. 67–73, 2016, doi: 10.1016/j.jsr.2015.12.002.

[7] N. S. A. Rudin, Y. M. Mustafah, Z. Z. Abidin, J. Cho, and H. F. M. Zaki, “Vision-based Lane Departure Warning System,” vol. 2, no. 2, pp. 166–176, 2018.

[8] G. Kaur and D. Kumar, “Lane Detection Techniques: A Review,” *Int. J. Comput. Appl.*, vol. 112, no. 10, pp. 975–8887, 2015.

[9] C. wang L. Zhang, F. Jiang, B. Kong, J. Yang, “real-time lane detection by using biologically inspired attention,” *Cognit. Comput.*, vol. 13, pp. 1333–1344, 2021.

[10] R. Zhang, Y. Wu, W. Gou, and J. Chen, “RS-Lane : A Robust Lane Detection Method Based on ResNeSt and Self-Attention Distillation for Challenging Traffic Situations,” *Hindawi*, pp. 1–12, 2021.

[11] F. Munir, S. Member, S. Azam, S. Member, and M. Jeon, “LDNet : End-to-End Lane Detection Approach using a Dynamic Vision Sensor,” *ARxIV:2009.08020V1 [cs.CV]*, pp. 1–10, 2020.

[12] S. Ghanem, P. Kanungo, G. Panda, S. C. Satapathy, and R. Sharma, “Lane detection under artificial colored light in tunnels and on highways: an IoT-based framework for smart city infrastructure,” *Complex Intell. Syst.*, 2021, doi: 10.1007/s40747-021-00381-2.

[13] D. Moher, A. Liberati, J. Tetzlaff, D. G. Altman, and P. Group, “Preferred reporting items for systematic reviews and meta-analyses : the PRISMA statement,” *PLoS Med.*, vol. 6, no. 7, p. Public Library of Science, e1000097, 2009, doi: 10.1136/bmj.b2535.

[14] C. Lee and J. H. Moon, “Robust Lane Detection and Tracking for Real-Time Applications,” *IEEE Trans. Intell. Transp. Syst.*, vol. 1, pp. 1–6, 2018,

doi: 10.1109/TITS.2018.2791572.

[15] J. Li, F. Jiang, J. Yang, B. Kong, M. Gogate, and K. Dashtipour, "Lane-DeepLab: Lane semantic segmentation in automatic driving scenarios for high-definition maps," *Neurocomputing*, vol. 465, pp. 15–25, 2021, doi: 10.1016/j.neucom.2021.08.105.

[16] D. Kavitha and S. Ravikumar, "Designing an IoT based autonomous vehicle meant for detecting speed bumps and lanes on roads," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 7, pp. 7417–7426, 2021, doi: 10.1007/s12652-020-02419-8.

[17] N. M. Gohilot, A. Tigadi, and B. Chougula, "Detection of pedestrian, lane and traffic signal for vision based car navigation," *2021 2nd Int. Conf. Emerg. Technol. INCET 2021*, pp. 1–6, 2021, doi: 10.1109/INCET51464.2021.9456137.

[18] S. Swetha and P. Sivakumar, "SSLA Based Traffic Sign and Lane Detection for Autonomous cars," *Proc. - Int. Conf. Artif. Intell. Smart Syst. ICAIS 2021*, pp. 766–771, 2021, doi: 10.1109/ICAIS50930.2021.9396046.

[19] T. K. B. Akbari, J. Thiyagalingam, R. Lee, "A Multilane Tracking Algorithm Using IPDA with Intensity Feature," *Sensors*, pp. 1–25, 2021.

[20] C. P. S. Lu, Z. Luo, F. Gao, M. Liu, K. Chang, "A Fast and Robust Lane Detection Method Based on Semantic Segmentation and Optical Flow Estimation," *Sensors*, pp. 1–18, 2021.

[21] S. W. Baek, M. J. Kim, U. Suddamalla, A. Wong, B. H. Lee, and J. H. Kim, "Real-Time Lane Detection Based on Deep Learning," *J. Electr. Eng. Technol.*, 2021, doi: 10.1007/s42835-021-00902-6.

[22] H. Park, "Implementation of lane detection algorithm for self-driving vehicles using tensor flow," *Adv. Intell. Syst. Comput.*, vol. 773, pp. 438–447, 2019, doi: 10.1007/978-3-319-93554-6_42.

[23] R. Yousri, M. A. Elattar, and M. Saeed Darweesh, "A Deep Learning-Based Benchmarking Framework for Lane Segmentation in the Complex and Dynamic Road Scenes," *IEEE Access*, vol. 9, pp. 117565–117580, 2021, doi: 10.1109/ACCESS.2021.3106377.

[24] A. M. Alajlan and M. M. Almasri, *Automatic lane marking prediction using convolutional neural network and S-Shaped Binary Butterfly Optimization*. Springer US, 2021.

[25] N. Kanagaraj, D. Hicks, A. Goyal, S. Tiwari, and G. Singh, "Deep learning using computer vision in self driving cars for lane and traffic sign detection," *Int. J. Syst. Assur. Eng. Manag.*, vol. 12, no. 6, pp. 1011–1025, 2021, doi: 10.1007/s13198-021-01127-6.

[26] "An improved and low-complexity neural network model for curved lane detection of autonomous driving system," *Wasser und Abfall*, vol. 21, no. 4, pp. 61–61, 2019, doi: 10.1007/s35152-019-0027-x.

[27] Q. Huang and J. Liu, "Practical limitations of lane detection algorithm based on Hough transform in challenging scenarios," *Int. J. Adv. Robot. Syst.*, vol. 18, no. 2, pp. 1–13, 2021, doi: 10.1177/17298814211008752.

[28] P. Lu, C. Cui, S. Xu, H. Peng, and F. Wang, "SUPER: A Novel Lane Detection System," *IEEE Trans. Intell. Veh.*, pp. 1–9, 2021, doi: 10.1109/TIV.2021.3071593.

[29] Z. Wu, K. Qiu, T. Yuan, and H. Chen, "A method to keep autonomous vehicles steadily drive based on lane detection," *Int. J. Adv. Robot. Syst.*, vol. 18, no. 2, pp. 1–11, 2021, doi: 10.1177/17298814211002974.

[30] C. Matching and B. Wu, "Lane Detection Using Sliding Window for Intelligent Ground Vehicle Challenge February," *Innov. Data Commun. Technol. Appl.*, pp. 871–881, 2021, doi: 10.1007/978-981-15-9651-3.

[31] M. Liu, X. Deng, Z. Lei, C. Jiang, and C. Piao, "Autonomous Lane Keeping System: Lane Detection, Tracking and Control on Embedded System," *J. Electr. Eng. Technol.*, vol. 16, no. 1, pp. 569–578, 2021, doi: 10.1007/s42835-020-00570-y.

[32] L. Zhang, F. Jiang, B. Kong, J. Yang, and C. Wang, "Real-Time Lane Detection by Using Biologically Inspired Attention Mechanism to Learn Contextual Information," *Cognit. Comput.*, vol. 13, no. 5, pp. 1333–1344, 2021, doi: 10.1007/s12559-021-09935-5.

[33] O. Rastogi, "Color masking method for variable luminosity in videos with application in lane detection systems," *Proc. Int. Conf. Mach. Intell. Data Sci. Appl.*, pp. 275–284, 2021, doi: 10.1007/978-981-33-4087-9.

[34] S. P. S. Deepak Dewangan, "Lane Detection for Intelligent Vehicle System Using Image Processing Techniques," *Data Sci.*, pp. 329–348, 2021, doi: 10.1016/s0300-7073(10)70732-5.

[35] J. Gong, T. Chen, and Y. Zhang, "Complex lane detection based on dynamic constraint of the double threshold," *Multimed. Tools Appl.*, vol. 80, no. 18, pp. 27095–27113, 2021, doi: 10.1007/s11042-021-10978-x.

[36] R. Muthalagu, A. Bolimera, and V. Kalaichelvi, "Vehicle lane markings segmentation and keypoint determination using deep convolutional neural networks," *Multimed. Tools Appl.*, vol. 80, no. 7, pp. 11201–11215, 2021, doi: 10.1007/s11042-020-10248-2.

[37] K. Y. Ren, H. C. Hou, S. Y. Li, and T. Y. Yue, "LaneDraw: Cascaded lane and its bifurcation detection with nested fusion," *Sci. China Technol.*

Sci., vol. 64, no. 6, pp. 1238–1249, 2021, doi: 10.1007/s11431-020-1702-2.

[38] D. K. Dewangan, S. P. Sahu, B. Sairam, and A. Agrawal, "VLDNet: Vision-based lane region detection network for intelligent vehicle system using semantic segmentation," *Computing*, vol. 103, no. 12, pp. 2867–2892, 2021, doi: 10.1007/s00607-021-00974-2.

[39] L. Review and L. L. Bachrach, "Lane and obstacle detection system based on single camera-based stereo vision system," *Appl. Adv. Comput. Syst. Proc. Int. Conf. Adv. Syst. Control Comput.*, pp. 259–266, 2021, doi: 10.1007/978-981-33-4862-2.

[40] A. N. Ahmed, S. Eckelmann, A. Anwar, T. Trautmann, and P. Hellinckx, "Lane Marking Detection Using LiDAR Sensor," *Lect. Notes Networks Syst.*, pp. 301–310, 2021, doi: 10.1007/978-3-030-61105-7_30.

[41] X. Y. Yan Wu, Fellin Liu, Wei Jiang, "Multi Spatial Convolution Block for Lane Lines Semantic Segmentation," *Lect. Notes Comput. Sci. B. Ser.*, vol. 12837, 2021, [Online]. Available: http://arxiv.org/abs/1704.06857.

[42] C. Y. Lee, J. G. Shon, and J. S. Park, "An edge detection–based eGAN model for connectivity in ambient intelligence environments," *J. Ambient Intell. Humaniz. Comput.*, no. Goodfellow 2014, 2021, doi: 10.1007/s12652-021-03261-2.

[43] "LLNet: A Lightweight Lane Line Detection Network," *Lu Zhang, Bin Kong, Can Wang*, 2021, doi: 10.1007/s42410-021-0508-x.

[44] J. N. Peng, Yong Qin, Jiansheng Hanxiao Zhang, "Lane Recognition System for Machine Vision," *10th Int. Conf. Comput. Eng. Networks*, pp. 388–398, 2020, doi: 10.1007/978-981-15-8462-6.

[45] A. Kasmi, J. Laconte, R. Aufrere, R. Theodose, D. Denis, and R. Chapuis, "An Information Driven Approach for Ego-Lane Detection Using Lidar and OpenStreetMap," *16th IEEE Int. Conf. Control. Autom. Robot. Vision, ICARCV 2020*, no. October, pp. 522–528, 2020, doi: 10.1109/ICARCV50220.2020.9305388.

[46] M. Fakhfakh, L. Chaari, and N. Fakhfakh, "Bayesian curved lane estimation for autonomous driving," *J. Ambient Intell. Humaniz. Comput.*, vol. 11, no. 10, pp. 4133–4143, 2020, doi: 10.1007/s12652-020-01688-7.

[47] E. S. Dawam and X. Feng, "Smart City Lane Detection for Autonomous Vehicle," *Proc. - IEEE 18th Int. Conf. Dependable, Auton. Secur. Comput. IEEE 18th Int. Conf. Pervasive Intell. Comput. IEEE 6th Int. Conf. Cloud Big Data Comput. IEEE 5th Cybe*, pp. 334–338, 2020, doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00065.

[48] K. C. Bhupathi and H. Ferdowsi, "An Augmented Sliding Window Technique to Improve Detection of Curved Lanes in Autonomous Vehicles," *IEEE Int. Conf. Electro Inf. Technol.*, pp. 522–527, 2020, doi: 10.1109/EIT48999.2020.9208278.

[49] Raja Muthalagu Anudeepsekhar Bolimera Kalaichelvi, "Lane detection technique based on perspective transformation and histogram analysis for self-driving cars," *Comput. Electr. Eng.*, pp. 1–16, 2020, doi: 10.1080/15551020902995363.

[50] A. Evlampev, I. Shapovalov, and S. Gafurov, "Map relative localization based on road lane matching with Iterative Closest Point algorithm," *PervasiveHealth Pervasive Comput. Technol. Healthc.*, pp. 232–236, 2020, doi: 10.1145/3430199.3430229.

[51] G. Zhang, C. Yan, and J. Wang, "Quality-guided lane detection by deeply modeling sophisticated traffic context," *Signal Process. Image Commun.*, vol. 84, no. January, p. 115811, 2020, doi: 10.1016/j.image.2020.115811.

[52] B. Dorj, S. Hossain, and D. J. Lee, "Highly curved lane detection algorithms based on Kalman filter," *Appl. Sci.*, vol. 10, no. 7, pp. 1–22, 2020, doi: 10.3390/app10072372.

[53] J. Z. and C. F. J. Hu, S. Xiong, "Lane detection and trajectory tracking control of autonomous vehicle based on model predictive control," *Int. J. …*, vol. 21, no. 2, pp. 285–295, 2020, doi: 10.1007/s12239.

[54] D. Liu, Y. Wang, T. Chen, and E. T. Matson, "Accurate Lane Detection for Self-Driving Cars: An Approach Based on Color Filter Adjustment and K-Means Clustering Filter," *Int. J. Semant. Comput.*, vol. 14, no. 1, pp. 153–168, 2020, doi: 10.1142/S1793351X20500038.

[55] S. Pansambal and U. Kumar, "A novel region-based iterative seed method for the detection of multiple lanes Suvarna Shirke & Ramanathan Udayakumar," *Int. J. Image Data Fusion ISSN*, pp. 1–20, 2019, doi: 10.1080/19479832.2019.1683623.

[56] Z. M. Chng, J. M. H. Lew, and J. A. Lee, "Roneld: Robust neural network output enhancement for active lane detection," *Proc. - Int. Conf. Pattern Recognit.*, pp. 6842–6849, 2020, doi: 10.1109/ICPR48806.2021.9412572.

[57] N. S. Rajat Agrawal, "Lane detection and collision prevention system for automated vehicles," *Appl. Comput. Vis. Image Process. Abstr.*, 2021, doi: 10.1007/s35145-020-0127-5.

[58] Z. Wang, W. Ren, and Q. Qiu, "LaneNet: Real-Time Lane Detection Networks for Autonomous Driving," *arXiv:1807.01726v1[cs.CV]*, 2018, [Online]. Available: http://arxiv.org/abs/1807.01726.

[59] F. Pizzati, M. Allodi, A. Barrera, and F. García,

"Lane Detection and Classification Using Cascaded CNNs," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, pp. 95–103, 2020, doi: 10.1007/978-3-030-45096-0_12.

[60] J. Z. Chen, Shizeng, Bijun li, Yuan Guo, "Lane detection based on histogram of oriented vanishing points," *Pattern Recognit.*, pp. 3–11, 2020, doi: 10.1007/978-981-15-3651-9.

[61] N. Ma, G. Pang, X. Shi, and Y. Zhai, "An All-Weather Lane Detection System Based on Simulation Interaction Platform," *IEEE Access*, vol. 8, pp. 46121–46130, 2020, doi: 10.1109/ACCESS.2018.2885568.

[62] C. Hasabnis, S. Dhaygude, and S. Ruikar, "Real-Time Lane Detection for Autonomous Vehicle Using Video Processing," *Lect. Notes Networks Syst.*, pp. 217–225, 2020, doi: 10.1007/978-981-15-0630-7_21.

[63] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang, "Robust lane detection from continuous driving scenes using deep neural networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 41–54, 2020, doi: 10.1109/TVT.2019.2949603.

[64] T. Andersson, A. Kihlberg, A. Sundström, and N. Xiong, "Road Boundary Detection Using Ant Colony Optimization Algorithm," *Adv. Intell. Syst. Comput.*, vol. 1074, pp. 409–416, 2020, doi: 10.1007/978-3-030-32456-8_44.

[65] D. L. Sabir Hossain, Oualid Doukhi, Inseung Lee, "Real-Time Lane Detection and Extreme Learning Machine Based Tracking Control for Intelligent Self-driving Vehicle," *Intell. Syst. Appl.*, pp. 41–50, 2020, doi: 10.1007/978-3-030-29513-4.

[66] V. T. Luu, V. C. Huynh, V. H. Tran, T. H. Nguyen, and T. N. H. Phu, "Traditional Method Meets Deep Learning in an Adaptive Lane and Obstacle Detection System," *Proc. 2020 5th Int. Conf. Green Technol. Sustain. Dev. GTSD 2020*, pp. 148–152, 2020, doi: 10.1109/GTSD50082.2020.9303108.

[67] S. S. R. Udayakumar, "Fusion model based on entropy by using optimized DCNN and iterative seed for multilane detection," *Evol. Intell.*, 2020, doi: 10.1007/s12065-020-00480-y.

[68] I. J. P. B. Franco, T. T. Ribeiro, and A. G. S. Conceição, "A Novel Visual Lane Line Detection System for a NMPC-based Path Following Control Scheme," *J. Intell. Robot. Syst. Theory Appl.*, vol. 101, no. 1, 2021, doi: 10.1007/s10846-020-01278-x.

[69] L. Review and L. L. Bachrach, "Driveable area detection using semantic segmentation deep neural network," *Comput. Intell. Data Sci.*, pp. 222–230, 2020, doi: 10.1007/978-3-030-63467-4.

[70] R. Pihlak and A. Riid, "Simultaneous road edge and road surface markings detection using convolutional neural networks," *Commun. Comput. Inf. Sci.*, pp. 109–121, 2020, doi: 10.1007/978-3-030-57672-1_9.

[71] D. R. and V. M. F. Santos, "Detection of Road Limits Using Gradients of the Accumulated Point Cloud Density," *Robot 2019 Fourth Iber. Robot. Conf.*, pp. 267–279, 2020, doi: 10.1007/978-3-030-35990-4.

[72] Y. Sun, L. Wang, Y. Chen, and M. Liu, "Accurate lane detection with atrous convolution and spatial pyramid pooling for autonomous driving," *IEEE Int. Conf. Robot. Biomimetics, ROBIO 2019*, pp. 642–647, 2019, doi: 10.1109/ROBIO49542.2019.8961705.

[73] J. Liu, "Learning full-reference quality-guided discriminative gradient cues for lane detection based on neural networks," *J. Vis. Commun. Image Represent.*, vol. 65, pp. 1–7, 2019, doi: 10.1016/j.jvcir.2019.102675.

[74] H. Zhan and L. Chen, "Lane Detection Image Processing Algorithm based on FPGA for Intelligent Vehicle," *Proc. - 2019 Chinese Autom. Congr. CAC 2019*, pp. 1190–1196, 2019, doi: 10.1109/CAC48633.2019.8996283.

[75] S. Srivastava and R. Maiti, "Multi-lane Detection Robust to Complex Illumination Variations and Noise Sources," *2019 IEEE Int. Conf. Electr. Control Instrum. Eng. ICECIE 2019 - Proc.*, 2019, doi: 10.1109/ICECIE47765.2019.8974796.

[76] D. Chang *et al.*, "Multi-lane Detection Using Instance Segmentation and Attentive Voting," *Int. Conf. Control. Autom. Syst.*, pp. 1538–1542, 2019, doi: 10.23919/ICCAS47443.2019.8971488.

[77] X. Jiao, D. Yang, K. Jiang, C. Yu, T. Wen, and R. Yan, "Real-time lane detection and tracking for autonomous vehicle applications," *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, vol. 233, no. 9, pp. 2301–2311, 2019, doi: 10.1177/0954407019866989.

[78] N. Kemsaram, A. Das, and G. Dubbelman, "An integrated framework for autonomous driving: Object detection, lane detection, and free space detection," *Proc. 3rd World Conf. Smart Trends Syst. Secur. Sustain. WorldS4 2019*, pp. 260–265, 2019, doi: 10.1109/WorldS4.2019.8904020.

[79] H. Bilal, B. Yin, J. Khan, L. Wang, J. Zhang, and A. Kumar, "Real-time lane detection and tracking for advanced driver assistance systems," *Chinese Control Conf. CCC*, pp. 6772–6777, 2019, doi: 10.23919/ChiCC.2019.8866334.

[80] L. A. Tran and M. H. Le, "Robust u-net-based road lane markings detection for autonomous driving," *Proc. 2019 Int. Conf. Syst. Sci. Eng. ICSSE 2019*, pp. 62–66, 2019, doi:

10.1109/ICSSE.2019.8823532.

[81] J. Philion, "FastDraw: Addressing the long tail of lane detection by adapting a sequential prediction network," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 11574–11583, 2019, doi: 10.1109/CVPR.2019.01185.

[82] A. A. Ali and H. A. Hussein, "Real-time lane markings recognition based on seed-fill algorithm," *PervasiveHealth Pervasive Comput. Technol. Healthc.*, pp. 190–195, 2019, doi: 10.1145/3321289.3321306.

[83] A. Yusuf and S. Alawneh, "GPU implementation for automatic lane tracking in self-driving cars," *SAE Tech. Pap.*, vol. 2019-April, no. April, pp. 1–5, 2019, doi: 10.4271/2019-01-0680.

[84] C. Y. Kuo, Y. R. Lu, and S. M. Yang, "On the image sensor processing for lane detection and control in vehicle lane keeping systems," *Sensors (Switzerland)*, vol. 19, no. 7, pp. 1–10, 2019, doi: 10.3390/s19071665.

[85] K. M. and P. Daniel, "Robust lane detection in hilly shadow roads using hybrid color feature," *2019 9th Annu. Inf. Technol. Electromechanical Eng. Microelectron. Conf.*, pp. 201–204, 2019.

[86] X. Pan and H. Ogai, "Fast lane detection based on deep convolutional neural network and automatic training data labeling," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E102A, no. 3, pp. 566–575, 2019, doi: 10.1587/transfun.E102.A.566.

[87] H. Liu and X. Li, "Sharp curve lane detection for autonomous driving," *Comput. Sci. Eng.*, vol. 21, no. 2, pp. 80–95, 2019, doi: 10.1109/MCSE.2018.2882700.

[88] Y. Son, E. S. Lee, D. Kum, and E. S. Lee, "Robust multi-lane detection and tracking using adaptive threshold and lane classification," *Mach. Vis. Appl.*, vol. 30, no. 1, pp. 111–124, 2019, doi: 10.1007/s00138-018-0977-0.

[89] B. S. S. Rathnayake and L. Ranathunga, "Lane detection and prediction under hazy situations for autonomous vehicle navigation," *18th Int. Conf. Adv. ICT Emerg. Reg. ICTer 2018 - Proc.*, pp. 99–106, 2019, doi: 10.1109/ICTER.8615458.

[90] Z. Feng, S. Zhang, M. Kunert, and W. Wiesbeck, "Applying neural networks with a high-resolution automotive radar for lane detection," *AmE 2019 Automot. meets Electron.*, pp. 8–13, 2019.

[91] W. Farag and Z. Saleh, "An advanced road-lanes finding scheme for self-driving cars," *2nd Smart Cities Symp. (SCS 2019), Bahrain, Bahrain*, pp. 1–6, 2019, doi: 10.1049/cp.2019.0221.

[92] H. Park, "Lane detection algorithm based on Hough transform for high-speed self driving vehicles," *Int. J. Web Grid Serv.*, vol. 15, no. 3, pp. 240–250, 2019, doi: 10.1504/IJWGS.2019.100835.

[93] K. S. Lee, S. W. Heo, and T. H. Park, "A lane detection and tracking method using image saturation and road width data," *J. Inst. Control. Robot. Syst.*, vol. 25, no. 5, pp. 476–483, 2019, doi: 10.5302/J.ICROS.2019.19.0008.

[94] H. P. B, "Robust Road Lane Detection for High Speed Driving of Autonomous Vehicles," *springer*, pp. 256–265, 2019, doi: 10.1007/978-3-030-15035-8.

[95] N. S. Parameswaran, E. Revathi Achan, V. Subhashree, and R. Manjusha, *Road detection by boundary extraction technique and hough transform*, vol. 30. Springer International Publishing, 2019.

[96] R. R. Dhanakshirur, P. Pillai, R. A. Tabib, U. Patil, and U. Mudenagudi, *A framework for lane prediction on unstructured roads*, vol. 968. Springer Singapore, 2019.

[97] A. Mahmoud *et al.*, "Real-time lane detection-based line segment detection," *2018 New Gener. CAS, NGCAS 2018*, pp. 57–61, 2018, doi: 10.1109/NGCAS.2018.8572124.

[98] Q. Li, J. Zhou, B. Li, Y. Guo, and J. Xiao, "Robust lane-detection method for low-speed environments," *Sensors (Switzerland)*, vol. 18, no. 12, pp. 1–18, 2018, doi: 10.3390/s18124274.

[99] W. Farag and Z. Saleh, "Road lane-lines detection in real-time for advanced driving assistance systems," *2018 Int. Conf. Innov. Intell. Informatics, Comput. Technol. 3ICT 2018*, 2018, doi: 10.1109/3ICT.2018.8855797.

[100] B. Li, Y. Guo, J. Zhou, Y. Cai, J. Xiao, and W. Zeng, "Lane Detection and Road Surface Reconstruction Based on Multiple Vanishing Point & Symposia *," *2018 IEEE Intell. Veh. Symp.*, no. Iv, pp. 209–214, 2018.

[101] E. Adali *et al.*, "Detecting road lanes under extreme conditions: A quantitative performance evaluation," *2018 6th Int. Conf. Control Eng. Inf. Technol. CEIT 2018*, pp. 1–7, 2018, doi: 10.1109/CEIT.2018.8751835.

[102] Y. Y. Ye, X. L. Hao, and H. J. Chen, "Lane detection method based on lane structural analysis and CNNs," *IET Intell. Transp. Syst.*, vol. 12, no. 6, pp. 513–520, 2018, doi: 10.1049/iet-its.2017.0143.

[103] Y. Y. Moon, Z. W. Geem, and G. T. Han, "Vanishing point detection for self-driving car using harmony search algorithm," *Swarm Evol. Comput.*, vol. 41, pp. 111–119, 2018, doi: 10.1016/j.swevo.2018.02.007.

[104] A. J. Humaidi, S. Hasan, and M. A. Fadhel, "FPGA-based lane-detection architecture for autonomous vehicles: A real-time design and development," *Asia Life Sci.*, no. 1, pp. 223–237,

2018.

[105] L. Xiong, Z. Deng, P. Zhang, and Z. Fu, "A 3D Estimation of Structural Road Surface Based on Lane-Line Information," *IFAC-PapersOnLine*, vol. 51, no. 31, pp. 778–783, 2018, doi: 10.1016/j.ifacol.2018.10.131.

[106] X. Chen and C. Luo, "Real-Time Lane Detection Based on a Light-Weight Model in the Wild," *2021 IEEE 4th Int. Conf. Comput. Commun. Eng. Technol. CCET 2021*, pp. 36–40, 2021, doi: 10.1109/CCET52649.2021.9544226.

[107] S. Liu, L. Lu, X. Zhong, and J. Zeng, "Effective road lane detection and tracking method using line segment detector," *Chinese Control Conf. CCC*, pp. 5222–5227, 2018, doi: 10.23919/ChiCC.2018.8482552.

[108] M. C. Olgun, Z. Baytar, K. M. Akpolat, and O. Koray Sahingoz, "Autonomous vehicle control for lane and vehicle tracking by using deep learning via vision," *2018 6th Int. Conf. Control Eng. Inf. Technol. CEIT 2018*, pp. 25–27, 2018, doi: 10.1109/CEIT.2018.8751764.

[109] Y. Huang *et al.*, "Spatial-Temporal Based Lane Detection Using Deep Learning," *14th IFIP Int. Conf. Artif. Intell. Appli- cations Innov.*, pp. 143–154, 2018.

[110] J. Xiao, L. Luo, Y. Yao, and W. Zou, "Lane Detection Based on Road Module and Extended Kalman Filter Lane Detection Based on Road Module and Extended Kalman Filter," *springer*, pp. 382–395, 2018, doi: 10.1007/978-3-319-75786-5.

[111] B. S. Khan, M. Hanafi, and S. Mashohor, "A Real Time Road Marking Detection System on Large Variability Road Images Database," *Lect. Notes Electr. Eng.*, vol. 488, pp. 31–41, 2018, doi: 10.1007/978-981-10-8276-4_4.

[112] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards End-to-End Lane Detection: an Instance Segmentation Approach," 2018, doi: 10.1109/CRV.2016.27.

[113] Y. Zhang, J. Gao, and H. Zhou, "ImageNet Classification with Deep Convolutional Neural Networks," *PervasiveHealth Pervasive Comput. Technol. Healthc.*, pp. 145–151, 2020, doi: 10.1145/3383972.3383975.

[114] H. Zhang *et al.*, "ResNeSt: Split-Attention Networks," *ARxIV:2004.08955V2 [cs.CV]*, pp. 1–12, 2020, [Online]. Available: http://arxiv.org/abs/2004.08955.

[115] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, pp. 5987–5995, 2017, doi: 10.1109/CVPR.2017.634.

[116] H. Zhou, J. Zhang, J. Lei, S. Li, and D. Tu, "Image semantic segmentation based on FCN-CRF model," *2016 Int. Conf. Image, Vis. Comput. ICIVC 2016*, pp. 9–14, 2016, doi: 10.1109/ICIVC.2016.7571265.

[117] Y. Lu, Y. Chen, D. Zhao, and J. Chen, "Graph-FCN for Image Semantic Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, pp. 97–105, 2019, doi: 10.1007/978-3-030-22796-8_11.

[118] S. Zhang, A. E. Koubia, and K. A. K. Mohammed, "Traffic Lane Detection using FCN," *ARxIV:2004.08977V1 [cs.CV]*, pp. 1–7, 2020, [Online]. Available: http://arxiv.org/abs/2004.08977.

[119] N. J. Zakaria, H. Zamzuri, M. H. Ariff, M. I. Shapiai, S. A. Saruchi, and N. Hassan, "Fully convolutional neural network for Malaysian road lane detection," *Int. J. Eng. Technol.*, vol. 7, no. 4, pp. 152–155, 2018, doi: 10.14419/ijet.v7i4.11.20792.

[120] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017, doi: 10.1109/TPAMI.2016.2644615.

[121] H. Wu and B. Zhang, "A deep convolutional encoder-decoder neural network in assisting seismic horizon tracking," *arXiv Prepr. arXiv1804.06814*, 2018, [Online]. Available: http://arxiv.org/abs/1804.06814.

[122] X. Ou *et al.*, "Moving Object Detection Method via ResNet-18 with Encoder-Decoder Structure in Complex Scenes," *IEEE Access*, vol. 7, pp. 108152–108160, 2019, doi: 10.1109/ACCESS.2019.2931922.

[123] V. Badrinarayanan, A. Handa, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling," 2015, doi: 10.1103/PhysRevX.5.041024.

[124] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation," *ARxIV:1606.02147V1 [cs.CV]*, pp. 1–10, 2016, [Online]. Available: http://arxiv.org/abs/1606.02147.

[125] W. Weng and X. Zhu, "UNet: Convolutional Networks for Biomedical Image Segmentation," *IEEE Access*, vol. 9, pp. 16591–16603, 2021, doi: 10.1109/ACCESS.2021.3053408.

[126] B. De Brabandere, D. Neven, and L. Van Gool, "Semantic Instance Segmentation with a Discriminative Loss Function," *ARxIV:1708.02551V1 [cs.CV]*, pp. 1–10, 2017, [Online]. Available:

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

http://arxiv.org/abs/1708.02551.

[127] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang, "Robust Lane Detection from Continuous Driving Scenes Using Deep Neural Networks," *ARxIV:1903.02193V2 [cs.CV]*, pp. 1–14, 2019.

[128] C. M. Kang, S. H. Lee, S. C. Kee, and C. C. Chung, "Kinematics-based Fault-tolerant Techniques: Lane Prediction for an Autonomous Lane Keeping System," *Int. J. Control. Autom. Syst.*, vol. 16, no. 3, pp. 1293–1302, 2018, doi: 10.1007/s12555-017-0449-8.

[129] A. Hata and D. Wolf, "Road marking detection using LIDAR reflective intensity data and its application to vehicle localization," *2014 17th IEEE Int. Conf. Intell. Transp. Syst. ITSC 2014*, pp. 584–589, 2014, doi: 10.1109/ITSC.2014.6957753.

[130] L. Frommberger and D. Wolter, "Structural knowledge transfer by spatial abstraction for reinforcement learning agents," *Adapt. Behav.*, vol. 18, no. 6, pp. 507–525, 2010, doi: 10.1177/1059712310391484.

[131] A. Kasmi, D. Denis, R. Aufrere, and R. Chapuis, "Map Matching and Lanes Number Estimation with Openstreetmap," *IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC*, pp. 2659–2664, 2018, doi: 10.1109/ITSC.2018.8569840.

[132] A. Kasmi *et al.*, "Probabilistic framework for ego-lane determination To cite this version : HAL Id : hal-03049401 Probabilistic framework for ego-lane determination," *2019 IEEE Intell. Veh. Symp.*, pp. 1746–1752, 2020.

[133] A. Joshi and M. R. James, "Generation of accurate lane-level maps from coarse prior maps and lidar," *IEEE Intell. Transp. Syst. Mag.*, vol. 7, no. 1, pp. 19–29, 2015, doi: 10.1109/MITS.2014.2364081.

[134] J. Fritsch, K. Tobias, and A. Geiger, "A New Performance Measure and Evaluation Benchmark for Road Detection Algorithms," *Intell. Transp. Syst.*, pp. 1693–1700, 2013.

[135] M. Aly, "Real time Detection of Lane Markers in Urban Streets," *arXiv:1411.7113v1 [cs.CV]*, 2014, doi: 10.1109/TITS.2017.2723575.

[136] M. Cordts *et al.*, "The Cityscapes Dataset for Semantic Urban Scene Understanding," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3213–3223, 2016, doi: 10.1109/CVPR.2016.350.

[137] J. Z. and C. F. J. Hu, S. Xiong, "Lane detection and trajectory tracking control of autonomous vehicle based on model predictive control," *Int. J. …*, vol. 21, no. 2, pp. 285–295, 2020, doi: 10.1007/s12239.

[138] X. Y. Yan Wu, Fellin Liu, Wei Jiang, "Multi Spatial Convolution Block for Lane Lines Semantic Segmentation," *Lect. Notes Comput. Sci. B. Ser.*, vol. 12837, 2021.

[139] C. Y. Lee, J. G. Shon, and J. S. Park, "An edge detection–based eGAN model for connectivity in ambient intelligence environments," *J. Ambient Intell. Humaniz. Comput.*, no. Goodfellow 2014, 2021, doi: 10.1007/s12652-021-03261-2.

[140] G. Zhang, C. Yan, and J. Wang, "Quality-guided lane detection by deeply modeling sophisticated traffic context," *Signal Process. Image Commun.*, 2020, doi: 10.1016/j.image.2020.115811.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3234442

NOOR JANNAH ZAKARIA received the Master of Philosophy (MPhil) and B. E. degree in Electronic System Engineering from the Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia. She is currently pursuing the Doctor of Philosophy (PhD) with the Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia. Artificial Intelligence, Supervised Learning, Deep Learning, Reinforcement Learning, and Object Detection in the Autonomous Vehicle field of interest are some of her research interests.

MOHD IBRAHIM SHAPIAI @ ABD. RAZAK is a senior lecturer and researcher at Universiti Teknologi Malaysia's Center of Artificial Intelligence and Robotics (CAIRO). He obtained his M Eng from the University of York in the United Kingdom in 2007 and his PhD in machine learning from Universiti Teknologi Malaysia in 2013. From March to April 2010, he worked as a visiting researcher at Waseda University's Graduate School of Information, Production, and Systems under Prof Dr Junzo Watada's supervision, and from June to July 2012, he worked at Leeds University's Faculty of Engineering under Prof Dr Vassili Toropov's supervision. Artificial Intelligence, Machine Learning, Brain-Computer Interface, and Swarm Intelligence are some of his research interests. In addition, he has been named a Certified NVIDIA Deep Learning Instructor.

RASLI ABD GHANI is a senior lecturer in the department of Electronic System Engineering from Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia (UTM).

MOHD NAJIB MOHD YASSIN (Member IEEE) In 2007, he got an M.Eng. in electronic engineering and a Ph.D. in electronic engineering from the University of Sheffield in the United Kingdom. He has been a Lecturer at the Universiti Malaysia Perlis' School of Microelectronics since 2013. Computational electromagnetics, conformal antennas, mutual coupling, wireless power transfer, array design, and dielectric resonator antennas are among his research interests.

MOHD ZAMRI IBRAHIM received both his B.Eng and M.Eng from Universiti Teknologi Malaysia, Malaysia and received his PhD from Loughborough University, United Kingdom. He is a Senior Lecturer from Faculty of Electrical and Electronics Engineering at University Malaysia Pahang, Malaysia. His research interests are in the area of computer vision, plasma science, embedded system programming, brain computer interaction, image processing, intelligent system and speech recognition.

NURBAITI WAHID received her BEng Electrical, MEng and Ph.D from Universiti Teknologi Malaysia. Currently, she is senior lecturer in Universiti Teknologi Mara, Malaysia. Vehicle dynamic systems, motion planning, and automated vehicle control systems are among her research interests.