Deep Learning in Lane Marking Detection: A Survey

Youcheng Zhang, Zongqing Lu*, Xuechen Zhang, Jing-Hao Xue, Qingmin Liao

Abstract—Lane marking detection is a fundamental but crucial step in intelligent driving systems. It can not only provide relevant road condition information to prevent lane departure but also assist vehicle positioning and forehead car detection. However, lane marking detection faces many challenges, including extreme lighting, missing lane markings, and obstacle obstructions. Recently, deep learning-based algorithms draw much attention in intelligent driving society because of their excellent performance. In this paper, we review deep learning methods for lane marking detection, focusing on their network structures and optimization objectives, the two key determinants of their success. Besides, we summarize existing lane-related datasets, evaluation criteria, and common data processing techniques. We also compare the detection performance and running time of various methods, and conclude with some current challenges and future trends for deep learning-based lane marking detection algorithm.

Index Terms—Lane marking detection, Traffic dataset, Deep network, Objective function, Evaluation metric.

I. Introduction

THE emergence of vehicles has liberated human feet from long journeys. In a modern society where almost every household has a car, people enjoy the convenience of travel, but also face many traffic hazards. To help make the driving safer and more intelligent, intelligent driving uses various sensing and detection equipment along with advanced algorithms to acquire, process and analyze the collected data promptly, so that driver can be alert to potential dangers and vehicles can be controlled automatically [1].

Various learning-based methods are widely used for their flexibility of handling complex scenes in modern intelligent transport systems [2], [3], and recent deep learning techniques further add wings to the development of intelligent driving. They gain immense success in reinforcement learning, unsupervised learning, supervised learning and their hybrids [4]. Various types of deep learning algorithms play their advantages in all aspects of intelligent driving. Generally speaking, deep reinforcement learning is the product of combining deep learning and reinforcement learning [5], [6]. Deep unsupervised learning refers to solving various problems based on unlabeled training samples. It includes generative deep structures

[7], such as autoencoder [8], [9], deep belief networks [10]–[12] (including Boltzmann machines) and Generative Adversarial Networks (GAN) [13], [14]. Deep supervised learning with labeled samples has achieved great success on Euclidean data and sequential data with Convolution Neural Network (CNN) [15]–[17] and Recurrent Neural Network (RNN) [18], [19], respectively, and supervised learning directly on various non-Euclidean data structure with Bayesian deep learning also arouses much attention [20].

Intelligent driving consists of a series of systems, such as adaptive cruise control, lane departure warning system, and lane-keep assistance. As a top priority of these systems, the safety in intelligent driving needs to be addressed from various technical aspects, such as road detection, lane marking detection, vehicle detection, and collision avoidance [21]. Therefore, lane marking detection plays a crucial role in intelligent driving.

As a task similar to lane marking detection, lane detection places more concerns on the lane in which the vehicle is currently traveling [22]. In actual driving, it is not enough to detect only the current lane situation. Lane marking detection extends the detection range from the current lane to the entire field of view. Drivers can use the lane marking detection results to understand the wider driving environment, thus preparing for dangerous road conditions such as sharp turns that may occur soon. Lane marking detection is also the foundation of many intelligent driving missions. No matter it is to predict the vehicle's moving trajectories [23] or to detect front vehicles [24], [25], lane marking detection can always act as an aid. In addition to intelligent driving, lane marking detection also contributes to robot navigation [26] and provides auxiliary information for visually impaired people's motion [27].

Unlike generic targets, lane markings occupy only a small but widely distributed proportion of the scene. Interference due to poor illumination, occlusion and similar textures makes the already challenging lane marking detection more difficult. There have been many approaches to lane marking detection, which can be categorized into handcrafted feature-based methods and deep learning methods.

Before the advent of large lane marking detection datasets, lane marking detection relies mainly on handcrafted feature-based methods. [28], [29] summarize these techniques from three aspects: preprocessing, feature extraction and curve fitting. The purpose of preprocessing is to filter out unwanted interference in the image. Color space conversion and filters, such as grayscale conversion [30], YC_bC_r chromaticity space conversion [31], mean filter and directional controllable filter [32], are often used for preprocessing. To extract lane marking
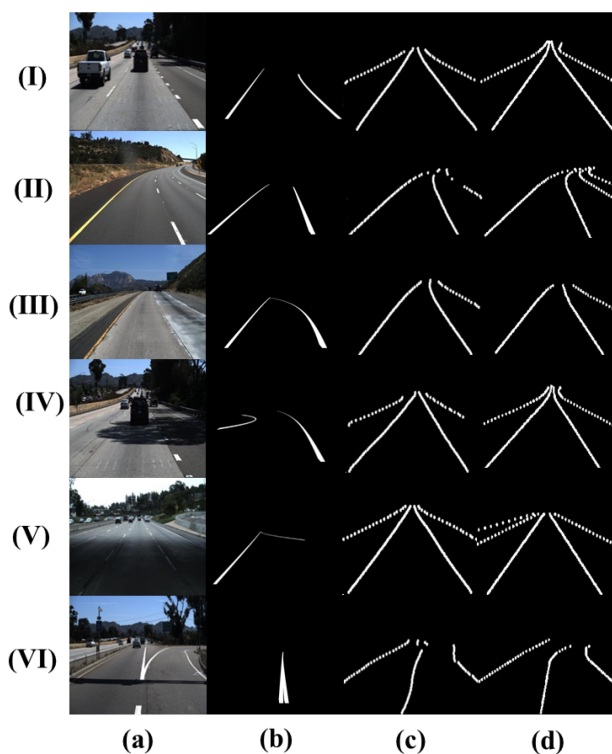
Fig. 1. Detection performance of different methods in various scenarios. Columns (different methods): (a) images; (b) results from a handcrafted feature-based method using color features, edge features and statistical histogram features; (c) results from Spatial Convolutional Neural Network (SCNN) [38]; (d) ground truth. Rows (different scenarios): (I) normal; (II) curve; (III) complex texture; (IV) shadow; (V) illumination; (VI) branch.

driven manner and has achieved superior performance. This has also led to the emergence of larger datasets of traffic scenes. Consequently, more and more deep learning methods are developed for lane marking detection. However, the early surveys of lane marking detection focus on handcrafted feature-based methods as well as related hardware platforms, such as [28], [29], and the reviews of deep learning methods specifically developed for lane marking detection are scarce. [39], [40] focus on the system construction with learning-based lane marking detection methods briefly introduced. For the success of deep learning methods, the designs of network structures and optimization objectives are two critical determinants. Therefore, in this paper, we review deep learning methods in lane marking detection by focusing on the network structures and optimization objectives. Different from [41], which studies the network structure of each algorithm, we also provide an in-depth analysis of the existing lane marking detection algorithms and explore the motivations behind the designs of different algorithms from the key problems that need to be solved in this field. A comparison of the mentioned surveys is summarized in Table I. Our survey covers the latest and representative deep learning techniques and deep neural networks for lane marking detection, including publications from journals, conferences and arXiv.

The rest of the survey is organized as follows. Section II summarizes the datasets used for lane marking detection. The commonly used optimization objectives are presented in Section III. The development of lane marking detection networks is elaborated in Section IV. Section V introduces data processing techniques, evaluation metrics and performance comparison of the methods. The discussion of challenges and future work in lane marking detection is presented in Section VI. Finally, conclusions are in Section VII.

II. LANE LINE DETECTION DATASETS

Current deep learning methods rely on a large amount of training data. With the advances of deep learning in intelligent driving, many datasets of traffic scenes have also emerged. In this section we introduce six major datasets for various intelligent driving tasks, then some datasets specifically for lane marking detection, and finally a summary and comparison of the characteristics of these datasets.

A. Traffic scene datasets

In addition to lane marking detection, vision-based autopilot includes many subtasks, such as traffic scene semantic segmentation, road sign detection and pedestrian detection. Several major traffic scene datasets are summarized below.

KITTI [69] is the largest dataset in autopilot scenarios, used to evaluate the performance in various computer vision tasks including optical flow, visual ranging, 3D objects detection and tracking. The real data collected in KITTI comes from urban, rural and highway scenes. In addition to as many as 15 vehicles and 30 pedestrians in each picture, there are various levels of occlusion and truncation. This dataset contains some scenes encountered during the actual driving process. There

features, many established operators are applied, for example, SIFT [31] and Canny [33], [34]. After the lane marking is initially detected based on the features, a series of fitting methods are used to refine the results. Commonly used fitting methods include Hough transform [34]–[36] and B-spline fitting [37]. These handcrafted feature-based methods have simple algorithms and fast detection. However, as shown in Fig. 1, handcrafted feature-based methods only achieve the desired performance when lane markings are obvious. When facing complex road conditions, these methods can no longer meet the requirements of accurate detection due to the lack of robustness of the features.

Compared with the methods using handcrafted features, deep learning methods can automatically extract the desired features that conform to the lane marking, obtaining better detection results in the face of occlusion and lane marking defects, etc. There are three main reasons for this: first, through learning from a large amount of data, deep learning can extract the features that are more conducive to achieving lane marking detection in various environments. Secondly, with sophisticated designs of network architectures, deep networks can extract effective features. Finally, by constructing optimization objective functions with appropriate constraints, deep networks can selectively learn the representation of relevant categories.

In more challenging scenarios, the deep learning method can extract discriminative features of lane markings in a data-

TABLE I

EXISTING SURVEYS RELATED TO LANE MARKING DETECTION. LPY DENOTES THE LATEST PUBLICATION YEAR OF COVERED PAPERS.

Survey (Publication Year)	Number of Covered Papers (LPY)	Key Words	Remarks
[29] (2013)	186 (2012)	Detection systems	Discussed handcrafted based methods and related hardware platforms
[28] (2014)	65 (2010)	Functional building blocks	Focused on sensing modalities and full detection system
[39] (2018)	112 (2017)	Algorithms, integration and assessment	Briefly introduced the lane marking detection algorithms based on deep learning
[40] (2018)	91 (2017)	Lane departure warning	Briefly introduced the lane marking detection algorithms based on deep learning
[41] (2020)	96 (2019)	Lane marking detection	Sorted algorithms according to network structure
ours (2021)	**114 (2021)**	**Lane marking detection**	**Comprehensive survey and in-depth analysis of lane marking detection algorithms, also including key problems, datasets and evaluation methods in this field**

TABLE II

CONTENT COMPARISON OF LANE MARKING DATASETS.

	KITTI road	BDD100K	CityScape	ApolloScape	Mapillary	CamVid	Caltech Lanes	Tusimple	CULane	VPGNet	LLAMAS	CurveLanes	DET
Number of pictures (frames)	579	100,000	5,000	143,906	19,035	182	1,225	6,498	133,235	21,097	100,042	150,000	5424
Year	2013	2018	2016	2018	2017	2008	2008	2017	2017	2017	2019	2020	2020
Multiple scenes	yes	yes	no	no	yes	no	no	no	no	no	no	yes	yes
Multiple cities	no	yes	no	no	yes	no	no	no	no	no	no	no	no
Multiple weather	no	yes	no	yes	yes	no	no	no	yes	yes	yes	yes	no
Multiple moments	no	yes	no	no	yes	no	no	no	yes	yes	yes	yes	no
Multiple line types	yes	yes	no	no	no	no	no	no	no	yes	yes	yes	yes
Lane marking labeling	yes	yes	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Citation (in this survey)	[45] [46] [57] [50]	[42] [43] [58] [52]	none	[47]	none	[48]	[49] [50]	[38] [42] [43] [44] [51] [50] [52] [53] [54] [59] [60] [61] [62] [63] [65] [56] [66] [64] [67]	[38] [42] [43] [49] [50] [52] [55] [47] [61] [62] [64] [66] [67] [65] [68]	[55]	[56] [47]	[51]	none

TABLE III

SETTING COMPARISON OF LANE MARKING DATASETS.

	KITTI road	BDD100K	CityScape	ApolloScape	Mapillary	CamVid	Caltech Lanes	Tusimple	CULane	VPGNet	LLAMAS	CurveLanes	DET
Collection Device	Grayscale Cameras Color Cameras Laser Scanner	Color Cameras	Color Cameras	Color Cameras + Laser Scanner	Color Cameras	Color Cameras	Color Cameras	Color Cameras	Color Cameras	Color Cameras	Color Cameras + LiDAR Maps	Dynamic Color Cameras Sensor	Vision
Resolution	1242×375	1280×720	2048×1024	3384×2710	900million(average)	960×720	640×480	1280×720	1640×590	640×480	1276×717	2560×1440	1280×800
Annotation	Pixel level + Rectangle coordinates	Key point coordinates	Pixel level	Pixel level	Pixel level	Pixel level	Key point coordinates	Key point coordinates	Key point coordinates	Key point coordinates	Pixel level	Key point coordinates	Pixel level

are some lane markings in KITTI, but the segmentation labels for the lane lines are missing.

BDD100K [70] is the largest and most diverse open driving dataset, containing 100,000 videos of more than 100 million frames. By taking keyframe samples for the 10th second of each video, 100,000 images are obtained and labeled. Marked content includes road target, road target classes, travelable area and lanes of multiple cities. Lane marking data in a variety of roads and lighting conditions are covered.

CityScape [71] is a semantic segmentation dataset that focuses on urban streetscape understanding. It contains various street views from 50 cities in different seasons. CityScape has a wealth of metadata (front and rear video frames, stereo, GPS and vehicle odometers), which can be used for pixel-level, instance-level and panoramic semantic segmentation. In the training set, 5,000 images are with high-quality annotations, and the rest 20,000 images are coarsely annotated. But there is no specific label for the lane marking.

ApolloScape[1] uses a mobile LiDAR scanner to collect point clouds from Reigl, which produces a precise and dense point cloud, making this dataset more accurate than KITTI, CityScape and BDD100K. ApolloScape was collected from four regions of two cities in China, providing streetscape images in a variety of weather conditions during the day. The traffic conditions and environment in the images are complicated. In addition to the lane markings, the dataset also includes images segmented by semantics such as perception, simulation scenes and road network data.

Mapillary[2] contains 25,000 high-resolution streetscape images covering a variety of weather conditions (sun, rain, snow,

fog, haze) and all-day lighting changes (dawn, day, dusk, night). Its annotations are five times finer than CityScape and contain markings for lane markings.

CamVid [72] is the first video collection with object-class semantic tags with metadata. It provides pixel-level lane marking annotations, but with only a few hundred images, very small compared with the databases aforementioned.

B. Datasets for lane marking detection

Sometimes general traffic scene datasets cannot meet the needs of lane marking detection. Therefore many datasets have been specifically constructed for lane marking detection.

Caltech Lanes[3] includes four clips taken on streets near Pasadena, California at different times of the day. This dataset is an early dataset for lane marking detection. Its image resolution is low and the magnitude is not large.

Tusimple[4] is larger and with higher image resolution compared with the Caltech dataset. Its content comes from the driving situation on the highway, including different levels of occlusion, different types of lane markings and different road conditions. It contains different traffic conditions throughout the daytime in moderate weather. The difficulty level of detection is general. There are 3,626 sequences of images. Each sequence contains 20 consecutive frames collected in one second and the 20th is labeled with the lane marking ground truth.

CULane [38] includes traffic conditions in Beijing at different times of the day, and its size is 20 times that of the Tusimple dataset. In addition to having different weather

---

[1]http://apolloscape.auto/index.html
[2]https://research.mapillary.com/

[3]http://www.vision.caltech.edu/malaa/datasets/caltech-lanes/
[4]https://github.com/TuSimple/tusimple-benchmark/issues/3

conditions and lighting levels, there are eight challenging lane marking detection scenarios, such as traffic congestion, shadow occlusion, lane marking missing and curved lane lines. This is currently the largest and challenging dataset specifically for lane marking detection.

VPGNet [55] was associated with the methods proposed to use the vanishing point to predict the lane marking. In addition to vanishing point labels, various types of lane markings and road signs are marked in detail. This dataset includes varying degrees of rainfall and nighttime images, which is challenging due to severe weather and extreme lighting conditions.

LLAMAS [73] is an unsupervised labeled lane marking dataset using automatically created maps to project markers into the image space, and it relies on sample-based optimization to improve the accuracy of the label. Unlike other datasets, the number of pixels marked on each lane marking is small and varies with the distance and position of markings, making LLAMAS more challenging and realistic.

CurveLanes [73] has more than 90% images containing curve lane lines, which compensates for the lack of curve scenes in previous datasets.

DET [74] uses Dynamic Vision Sensor (DVS) for traffic scenes data collection, in which the data has low latency and high dynamic range. Due to the characteristics of DVS, images exclude the effects of illumination changing and redundant background (sky, road surface, etc.). However, the application scope of this dataset is limited since most vehicles are equipped with color cameras instead of DVS.

C. Dataset summary

To facilitate the reader to select appropriate datasets according to different task requirements, we summarize the characteristics of the above-mentioned datasets and their citation in Table II, with more technical details in Table III.

Most of the existing traffic scene datasets are collected in actual driving scenes and contain a variety of complex situations, such as complex lane marking distribution, incomplete lane marking information, road texture interference and shadow occlusion. However, the lane marking detection datasets under extreme weather conditions are very scarce. Although there are rainy traffic scenes in VPGNet, and CULane contains a large number of dark driving scenes, more challenging datasets, which integrate multiple complex traffic scenarios including extreme weather conditions and various visibility scenes, such as different levels of rain, snow and fog and more moments in a day (dusk, night, etc.), are expected.

III. REPRESENTATIVE OBJECTIVE FUNCTIONS

The design of the optimization objectives is critical to the success of deep learning methods. Well-behaved objective functions serve as the basis for many algorithmic improvements. In this section, we discuss the objective functions used for lane marking detection in different learning scenarios. Generally, learning algorithms are divided into two main categories: supervised learning and unsupervised learning. Classification and regression are two main themes of supervised learning. Therefore, we will discuss the involved objective

functions from three aspects: for classification, for regression and for effective unsupervised learning (via deep generative models). Illustrative summaries is shown in Fig. 2.

A. Objective functions for deep supervised learning models

1) Softmax based cross entropy and variants for classification: Many existing deep learning-based lane marking detection algorithms model the whole procedure of learning a pixel-wise classifier with well-annotated data, which indicates whether the present pixel can be categorized into lane marking given the local/global semantic information. Such a formulation is well studied in the deep learning community and softmax-based cross entropy and its variants are widely adopted as objective functions for optimizing these models. Given two C-dimensional discrete probability distributions $y(c)$ and $\widehat{y}(c)$, $c = 1, \ldots, C$, the cross entropy $H(y||\widehat{y})$ of the distribution q relative to p is defined as

$$H(y||\widehat{y}) = -\sum_{c=1}^{C} y(c) \log \widehat{y}(c). \quad (1)$$

When Eq.(1) is applied to classification, the support distribution y is often taken as C-dimension one-hot label vector y_{oh}, the output distribution $\widehat{y}$ of the neural network is obtained by the mentioned softmax layer as $\widehat{y}_{sftmx}$, and Eq.(1) becomes

$$H(y_{oh}||\widehat{y}_{sftmx}) = -\log \widehat{y}(j) = -\log \frac{e^{f_j(x)}}{\sum_{i=1}^{C} e^{f_i(x)}}, \quad (2)$$

where f_i is the feature mapping of the ith class and j denotes the ground truth class of input x.

Although the softmax based cross entropy has demonstrated remarkable performance in various classification tasks, many studies still focus on improving its performance to tackle some long-existing problems such as hard sample mining and class imbalance, and these variants can also be found in lane marking detection. For example, [44] further regularizes the intra-class distance of the samples in the same class; [58], [75], [76] allocate different weights to different classes to alleviate training problems brought by unbalanced classes.

2) Regression losses: When the output of a deep learner is expected to be continuous, regression losses are more suitable compared with those classification ones mentioned in section III-A1. In lane marking detection, the most commonly used regression losses are the mean squared error (MSE) [53], [59], [64], [75], [77]–[79], mean absolute error (MAE) [54], [79], [80], and Huber loss defined as

$$L_\delta(y, \widehat{y}) = \begin{cases} \frac{1}{2}(y - \widehat{y})^2, & |y - \widehat{y}| \leq \delta \\ \delta \cdot (|y - \widehat{y}| - \frac{1}{2}\delta), & \text{otherwise} \end{cases} \quad (3)$$

where y is the ground truth, $\widehat{y}$ is the predicted value and δ is a parameter.

It is well known that MSE and MAE have their own characters. Generally speaking, MSE focuses more on larger errors, while MAE focuses on smaller ones. In this sense, MAE tends to be more robust to outliers than MSE. Combining the merits of MSE and MAE, the Huber loss is often used in various deep learning-based regression problems. For more introductions on various robust regression losses and extensions in the deep learning area, readers can refer to [81].
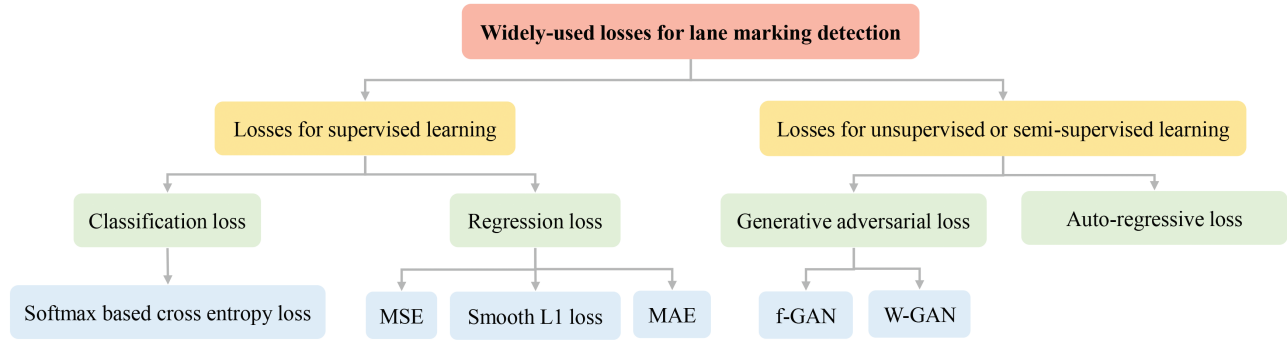
Fig. 2. Summary of widely-used objective functions for lane marking detection.

TABLE IV
COMPARISON WITH OTHER APPROACHES. x DENOTES INPUT IMAGE, y IS THE CORRESPONDING LABEL, $\widehat{y}$ IS THE PREDICTION RESULTS. THE REMAINING
VARIABLES ARE INTRODUCED IN THE RELEVANT PART OF SECTION IV.

Discriminative Loss (Loss for supervised learning)	Regression Loss	MSE	$L = \int_0^t (l_{\widehat{y}}(x) - l_y(x))^2 dx$ [59]	(1)
		MAE	$Loss = \sum_{i=1}^{15} |xp_i - xg_i| + \sum_{i=1}^{15} |yp_i - yg_i|$ [54]	(2)
		Huber	$L_\delta(y,\widehat{y}) = \begin{cases} \frac{1}{2}(y - \widehat{y})^2, & |y - \widehat{y}| \le \delta \\ \delta \cdot (|y - \widehat{y}| - \frac{1}{2}\delta), & \text{otherwise} \end{cases}$	(3)
		Hinge Loss	$L_{var} = \frac{1}{C}\sum_{c=1}^C \frac{1}{N_c}\sum_{i=1}^{N_c}[||\mu_c - x_i|| - \delta_v]_+^2$ $L_{dist} = \frac{1}{C(C-1)}\sum_{c_A=1}^C \sum_{c_B=1, c_A \ne c_B}^C [\delta_d - ||\mu_{c_A} - \mu_{c_B}||]_+^2$ [44]	(4)
	Cross Entropy		$L_{cls} = \sum_i^{N_b} \sum_n^N y_n^i \log \widehat{y}_n^i + \sigma(1 - y_n^i)\log(1 - \widehat{y}_n^i)$ [75]	(5)
Generative Loss (Loss for unsupervised/ semi-supervised learning)	GAN		$L_{gen}(x,y;\theta_{gen},\theta_{disc}) = L_{cel}(G(x;\theta_{gen});x,\theta_{disc})) + \lambda * L_{emb}(\widehat{y},y;x,\theta_{disc})$ $L_{disc}(x,y;\theta_{gen},\theta_{disc}) = L_{bce}(D(G(x;\theta_{gen});\theta_{disc}),1)) + L_{bce}(D(y;\theta_{disc}),0))$ [53]	(6)
	Auto Regressive Loss		$p(\mathbf{l}|\mathbf{x}) \approx p(\mathbf{l}_1|\mathbf{x})\prod_{i=1}^{c-1} p(\mathbf{l}_{i+1}|\mathbf{l}_i,\mathbf{x}) = p(\mathbf{l}_1|\mathbf{x})\prod_{i=1}^{c-1} p(\triangle w_i|\mathbf{l}_i,\mathbf{x})$ [61]	(7)

B. Objective functions for deep unsupervised / semi-supervised learning models

In recent years, deep unsupervised / semi-supervised learning algorithms have demonstrated great capability in many challenging learning cases, such as the absence of enough well-annotated training data and the pursuit of interpretable deep learning. Among the existing methods, autoregressive models [82], normalized flow models [83], variational autoencoder [84] and GAN may be the most widely-used ones for various computer vision tasks, and lane marking detection is not an exception. Compared with the objective functions of fully supervised manners, such as classification losses and regression losses, objective functions for deep unsupervised / semi-supervised models aim to acquire the intrinsic properties of the high-dimension data. For example, [61] share a similar methodology with the autoregressive models which model the potential data distribution sequentially, [53] uses the conditional generative adversarial network as regularization to generate realistic semantic segmentation. Among various deep generative models, GAN may be the most representative and its objective functions can be defined under the framework of integral probability metrics [85] as follows:

$$\inf_\theta \sup_{f \in \mathcal{F}} \mathbb{E}_{x \sim \mathbb{P}_r}[f(x)] - \mathbb{E}_{x \sim \mathbb{P}_\theta}[f(x)], \qquad (4)$$

where $\mathbb{P}_r$ is the probability of the real data, $\mathbb{P}_\theta$ is the probability of the output of the generator networks, and f is the discriminator network within the function class $\mathcal{F}$. The different choices of $\mathcal{F}$ may lead to different variants of GAN, such as f-GAN [86] and W-GAN [87]. Here we would like to point out that the current use of conditional deep generative models may be insufficient for fully exploiting the fruit of powerful deep generative models, and one potential research direction of deep generative models to lane marking detection is to synthesize high-quality training samples. More works on this emerging area are expected.

Examples of the above-mentioned objective functions applied in lane marking detection are shown in Table IV. To the best of our knowledge, there is no one-fits-all objective function to train neural networks for different purposes, and the context of application is of the key consideration on which a suitable choice is made. In this case, for different steps or constraints imposed for the same aim, it is very common to find in recent literature [47], [65]–[67] that more than one kind of objective function is adopted throughout the whole procedure. Because objective functions are among the most important factors affecting the performance of deep learning algorithms, more comprehensive reasons for the choice and more detailed experimental validity are expected for future research in the lane marking detection community.

IV. ADVANCES OF LANE MARKING DETECTION NETWORKS

Lane marking detection algorithms based on handcrafted features are feature-specific and often perform poorly on complex traffic scenes. In contrast, deep learning allows an algorithm to automatically learn feature representation at various levels, avoiding the limitation of manual feature extraction.

CNN is a neural network designed to process data (such as images) with a similar network structure and has excellent feature extraction capabilities and classification performance.

Vision-based intelligent driving tasks, including lane marking detection, usually use the traffic scene images (frames in the video) collected by the vehicle-mounted camera as the data source [29]. This is the reason why almost all lane marking detection algorithms are inseparable from CNN. Despite the success of basic CNN, the following problems and questions have inspired recent and more effective deep architectures for lane marking detection:

1) Different from ordinary semantic segmentation targets, lane marking has unique color (white, yellow) and shape (slender line). How to integrate these unique properties into the design of a neural network and improve its performance?

2) There are many types of lane markings, such as single solid lines and double yellow lines. Sometimes simply detecting the position of lane marking is not enough to meet the needs of traffic scene understanding. To better complete the assisted driving task according to the traffic regulations, how can a proposed algorithm be able to further determine the type of lane markings?

3) For assisted driving, real-time performance is an important consideration, as lane marking detection needs to be carried out while the vehicle is traveling. How can the efficiency of an algorithm be improved while maintaining detection accuracy? To make the algorithm suitable for vehicle-mounted scenarios, how to reduce the network magnitude?

According to the solutions to the above three questions, deep architectures in lane marking detection methods will be discussed in Sections IV-A, IV-B and IV-C, respectively.

A. Deep architecture focusing on lane marking structure

The advances of deep architectures for lane marking detection are inseparable from that of semantic segmentation network structures. Therefore, in this section, we will use the advances of semantic segmentation networks in recent years as a clue to start a discussion of lane marking detection networks.

[88] combines CNN and the random sample consensus (RANSAC) algorithm to refine lane marking detection results under relatively complex road conditions. Different from [89], which extracts a small image patch centered on each pixel as the input of CNN to predict the semantic label of the pixel, [88] manually extracts the Region of Interest (ROI) as the input of CNN, and directly outputs the lane marking candidates by using multilayer perceptron in the fully connected layer. To add more auxiliary information about lane markings, [90] inputs the left, center and right three-angle images obtained by the car camera into CNN. However, the shortcomings of these traditional CNN-based methods are obvious. First, it needs to traverse each pixel (in the image or ROI) to extract the patch for training and prediction, which is space-consuming, and there is a lot of redundant calculation between overlapped patches; Secondly, it is difficult to decide the appropriate size of a patch. Too small patches would lack the context information, while too large patches will increase calculation. Finally, the position information, though important for lane marking segmentation, would be discarded by the pooling layer in CNN.

To improve the computational efficiency in CNN and to maintain position information, [91] proposes Fully Convolutional Networks (FCN). FCN replaces the fully connected layer of traditional CNN with a convolutional layer. After the replacement, the output feature map is upsampled by deconvolution, so that the output prediction map and the input map have the same size and the position information can be retained. In this way, it is sufficient to directly calculate the classification loss between corresponding pixels during backpropagation, thereby achieving the pixel-level end-to-end semantic segmentation. Furthermore, the FCN avoids the disadvantage that the input image size in the fully connected layer must be fixed, that is, it can handle any size of input.

[92] applies FCN in lane marking detection, which allows the network model to obtain multiple input area scores in a single forward calculation and improves the efficiency. Given the linear characteristics of the lane marking, they added regression loss to the pixel-level loss function. However, the results obtained are relatively fuzzy and smooth due to deconvolution. Also, this algorithm classifies each pixel separately and does not fully consider the spatial consistency between pixels. To solve the loss of useful information during FCN upsampling (deconvolution) and to increase contextual contact, [93] proposes the U-Net for medical image segmentation, which achieves the best results on an ISBI challenge.

Compared with FCN, U-Net has three improvements: firstly, it is a symmetrical model. U-Net gradually reduces the spatial dimension of the input data by using the pooling layer of the encoder, while the decoder gradually recovers the details of the target and the corresponding spatial dimension through the network layer such as the deconvolution. The feature map between the symmetric paths is fused, allowing the network to propagate context information to higher resolution layers. Secondly, unlike the feature fusion method in FCN which adds points point by point, U-Net uses features that are stitched together in the channel dimension to form more *thick* features, which can better preserve the correlation between features. Thirdly, U-Net completely cancels the operation of full connection in FCN, which not only saves the number of parameters but also facilitates the fusion of high and low layer features. In this way, U-Net can train the model with a small amount of data to efficiently obtain higher segmentation accuracy. Many of the deep structures borrow the ideas of symmetric structure and long skip connection from U-Net for lane marking detection, such as [48], [53], [75].

However, the simple U-Net-based segmentation network does not take into account three unique properties of lane markings:

1) Lane markings have the spatial structure of slender linear. The convolution operation makes this structure destroyed, which reduces the detection efficiency.

2) Objects in the same scene have certain correlations. Therefore, networks should use the scene information a priori when segmenting lane markings.

3) There is a certain regularity between the positions of lane markings in the same scene. The network should make full use of this inherent constraint.

Inspired by the above three aspects, many researchers have combined the inherent characteristics of lane markings to improve the codec structure of U-net, which are discussed

in the following three parts: leverage spatial structure, scene prior and position regularity, respectively.

1) Leverage spatial structure: Leveraging the spatial structure of lane markings during network learning can greatly improve detection performance. The convolution mode and the objective distribution are two important components of neural networks and two starting points for modification.

Improvements in Convolution Mode.

To preserve the internal data structure and avoid downsampling which loses useful information, dilated convolution [94] was proposed. It injects holes into the standard convolution map, and thus increases the receptive field without losing information by pooling.

Many segmentation network structures, for example, E-Net [95] and EDANet [96], make use of dilated convolution, and some for lane marking detection, such as [43], [48], [49], [97]. As the lane marking pixel is of small size, narrow, and easily lost during downsampling, [97] proposes two techniques based on EDANet, which are Feature Size Selection (FSS) and Degressive Dilation Block (DD Block) to improve the lane marking segmentation accuracy. FSS is designed to extract fine features. In EDA-FSS, the EDA block is arranged after each downsampling block to extract multi-size features before the vanishing of thin lane features. As incremental dilation rates will lead to inconsistent spatial information in local features of deep layers, the DD Block of convolution layers with degressive dilation rates is designed and inserted into EDANet to keep spatial information consistent in layers of different depths.

However, there are some problems with dilated convolution: first, not all pixels are used for calculation when dilation rate is not continuous, so the continuity of object information would be lost; second, the design of dilated convolution is to obtain long-ranged information, but the large dilation rate may only be effective for some large objects, harmful for small objects. Hence, how to handle objects of different sizes is the key to designing a dilated convolution network.

For linear targets such as lane lines, [38] proposes SCNN focusing on making the internal connection of feature maps stronger. A new convolution method was designed to change the direction of the information flow, as shown in Fig. 3. Suppose the output of a convolutional layer is a feature map of size $C \times W \times H$. The feature map can be divided into H slices of size $C \times W$. A convolution kernel of the same size is applied to each slice in turn, with the middle slice convolved after the previous layer's convolution results are superimposed on it. The spatial convolution can be operated in four directions: up, down, left and right. Spatial convolution changes the direction of the information flow in the convolution, making the internal connection of the feature map stronger. Therefore, it is advantageous for target detection like lane lines and walls, which have long continuous shape structure.

To make the network more suitable for detecting long and narrow targets, SpinNet [68] designs a spinning convolution layer, in which 1D convolution is used instead of square convolution to obtain a narrow receptive field. Besides, feature maps will rotate by different angles in this convolution module to extract and fuse features in multiple directions.

Graph Convolutional Network (GCN) is another convolution design to strengthen target structure. It can use the adjacency matrix to define the connection relationship of each node, thereby highlighting the structure between the nodes. In the color-based segmentation network of [77], it uses the connectivity structure of the graph to keep the relationship between adjacent pixels and thus stress the lane marking structure.

Among the new convolution methods mentioned above, SCNN and SpinNet are specifically proposed for slender targets. However, it still needs to be improved in terms of calculation speed. Here we would like to point out that the current lane marking detection networks are mostly committed to improving the segmentation performance of the network. How to develop new convolution methods with fast segmentation of linear targets remains an open question for the future.

Improvements in Objective Distribution Learning.

Instead of learning pixel-level features as do traditional CNNs, GAN focuses on the learning at distributional level, consisting of two CNNs: one is called generator for producing results that approximate real data and the other is called discriminator for judging whether the generated data is true.

EL-GAN [53] leverages GAN in lane marking detection for more realistic and rich-structured semantic segmentation results, as shown in Fig. 4(a). The generator of EL-GAN uses a fully convolved U-Net structure with skip connections, while a DenseNet structure is applied in the discriminator. In order to highlight the pairing differences between the original image and different tags, EL-GAN adds another discriminator with shared parameters. The optimization objectives of EL-GAN thus contain that of the original GAN and an item that measures the pairing difference, as shown in Table IV (6). In order to handle complex traffic scenes, such as lane markings are obscured or defective, [98] designs Ripple-GAN blending the ideas of feature fusion, Wasserstein generative adversarial training and multi-target segmentation. Experiments showed that Ripple-GAN achieves excellent performance, especially when lane marking information is incomplete.

The attention mechanism weights different features differently in the feature maps to highlight the factors that are crucial to the objective distribution. [52] designs semantic-guided channel attention and a pyramid deformable convolution module to enhance the structural features of lane markings during the network training. [62] combines self-attention with channel attention to capture the global context and strengthen important features of lane marking. [43] uses self-attention with knowledge distillation to enhance the continuity of contextual information. This Self Attention Distillation (SAD) is trained without a teacher network, and the distillation is guided by the layer-wise attention knowledge extracted from the network itself, as shown in Fig. 4(b). In this way, lane marking detection would be substantially improved in challenging scenes without the extra complexity of the algorithm.

No matter the reform of the convolution mode or the emphasis of the objective distribution on the feature map, these researches successfully drive the semantic information of the target (lane marking) to be learned in the process of deep learning. In future research, it merits further investigation to
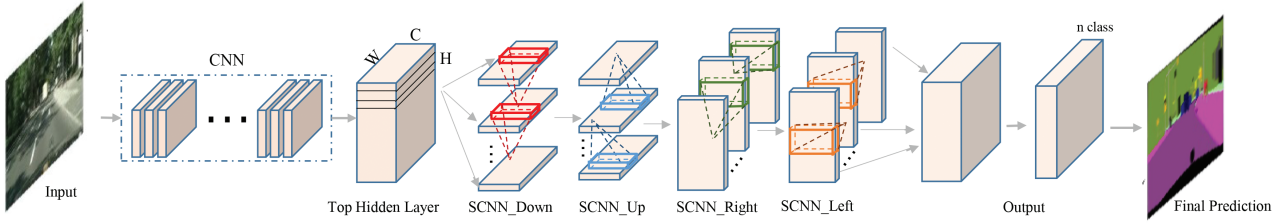
Fig. 3. The Overview of SCNN [38]. SCNN_Down, SCNN_Up, SCNN_Right and SCNN_Left are four types in SCNN according to the order of slice convolution.
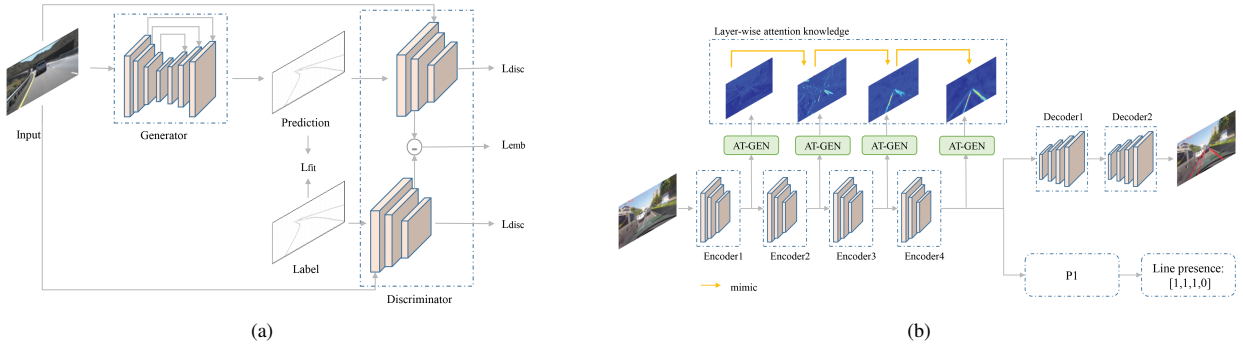

Fig. 4. Objective Distribution Constraint. (a) The overview of EL-GAN [53]. Adding the embedding loss to GAN to enhance the connection between the lane marking prediction results and the original scene. (b) Lane marking detection with SAD [43]. The SAD is used to extract the attention map to enrich the context information.

make such learning more interpretable, so as to provide more ideas for improving lane marking detection.

2) Leverage scene prior.: The information in the scene can supplement auxiliary to the segmentation of objectives. VGPNet [55] uses lane vanishing points to guide lane marking detection. In order to improve the algorithm versatility and reduce the difficulty of labeling, [99] proposes a new method of vanishing points-assisted lane detection based on heatmap regression. [100] first detects lane region mask and then calculates the convex hull surrounding the mask to attain lane marking. A idea similar to [100] is used in [58].

In order to introduce the global scene information prior to segmentation, PSPNet [101] proposes a pyramid pooling module to fuse features at different scales. Inspired by this, [49] proposes ZoomModule, which inputs the panoramic image and multiple patches from the original image into two CNNs to extract features and merges the obtained features of different scales. In this way, [49] can flexibly change the field of view to improve the detection of details.

Since the lane markings have a slender structure, [61] make use of the distribution of pixels around the lane markings in pixel-level prediction. It designs a FastDraw module to maximize the likelihood of polylines and assumes that the current lane marking point coordinates depend only on the previous c decoded coordinates, as shown in Table IV (7).

The feature pyramid simply fuses features of various sizes, which inevitably has redundant information. How to fuse these features more efficiently and effectively for lane markings requires further thinking.

3) Leverage position regularity.: The lane markings are different from other segmentation targets, and their distribution in the scene has a certain regularity. For example, the lane markings are all distributed on the road surface; the lane markings are parallel in the Bird's Eye View (BEV), and the lane markings in the continuous scenes have continuity. These regularities can help to infer the lane marking location based on the known lane markings, and thus the problem of the incomplete lane markings.

To predict the invisible lane markings, [102] manual reasons the unseen lane markings with the assumption that the distances between lanes are even and the lane marking shape of one lane is similar. Moreover, Yolo [103] and convolutional patch network with spatial prior [104] are employed to remove cars and non-road surface, respectively. In this way, lane marking features can be accurately extracted on the obstacles removed images.

In addition to manually determined assumptions for inferring lane markings, different network structures are proposed to use lane marking information in different view fields. [50] achieves more efficient vanishing lane marking prediction by deep learning with the assistance of an adaptive dark-light-dark method. [65] designs horizontal reduction modules to treat lane marking detection as a line-by-line classification problem, making use of the inherent shape of lane markings. Neural Architecture Search (NAS) is a technology for automatically designing high-performance network structures based on sampled data through algorithms. [51] uses NAS to capture lane marking's long-ranged consistency and short-ranged curve information, achieving a better detection performance of curve
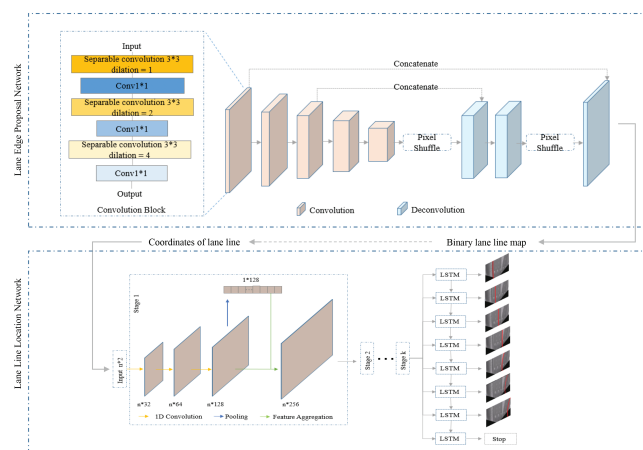
Fig. 5. The overview of LaneNet in [75], where LSTM is used for mutual reasoning of different lane markings in the same scene.

lanes. Such a method, however, is computationally expensive.

Long Short-Term Memory (LSTM) has been introduced to the lane marking detection to strengthen the reasoning of association. LSTM, a special type of RNN, can learn long-term dependency information in a sequence. For this reason, some researches combine it with the CNN-based network to exploit the correlation of lane markings within or between frames.

Lane edge proposal and location are respectively implemented by two networks in LaneNet [75]. LSTM is used at the end of lane marking location network to sequentially predict different lanes, as shown in Fig. 5. Lane edge proposal network based on encoder-decoder aims to produce binary lane line map. In the encoder, the combination of depthwise separable convolutions and 1×1 convolutions are used to reduce computation cost. Dilated convolution is used to enlarge the receptive field. In order to reduce the training complexity, sub-pixel convolution layers (used in [105]) and skip connections are employed to restore feature resolution in the decoder. The loss function of this network is weighted binary cross entropy, as shown in Table IV (5).

A lane line location network is proposed to detect lane markings in multiple patterns more accurately. Lane marking coordinates extracted from the binary lane marking map is entered in this network, so the size of the input is reduced and the prediction can be accelerated. In order to encode the coordinates into a low-dimensional overall representation, the location network applies a series of 1D convolution and pooling layers in the encoder. To solve the problem of an unknown number of lane markings, LSTM is applied in the decoder to sequentially infer lane markings. Each lane marking is fitted with a quadratic curve. To avoid manually labeling the ground truth of each curve, the distance between the input coordinates and the predicted curves is calculated in loss functions, which leads to weakly supervised training. However, the detection results can be unsatisfying when weak supervision is used as it depends too much on the results of the lane edge proposal network.

In order to obtain satisfying performance in challenging situations such as heavy shadow, lane mark degradation and occlusion, [76] incorporates the information from continuous frames to infer lane markings. Different from [75], a ConvL-STM block is inserted in the middle of the codec to fuse the features extracted from frames at different moments and retain essential information to predict lane markings. [56] uses a similar framework to [76] and inserts two convolutional gated recurrent units to speed up LSTM.

To deal with the situation of scene changes, mutual inference within frames is more effective than inter-frame inference; however, LSTM-based reasoning needs one frame to be repeatedly calculated so that all lane markings are detected. In addition to using LSTM to explore the position regularity of lane markings, [63] uses optical flow estimation between frames to obtain the spatio-temporal features of the image and assist lane marking detection. Some information that is more stable and helpful to efficiently infer lane markings needs to be mined.

B. Deep architecture focusing on lane marking classification

To better understand the traffic scene, the lane marking detection algorithm is expected to determine the type of lane marking while accurately detecting its position, which can provide more useful information for assisted driving. [45] inputs the lane markings detected by handcrafted feature-based method into a CNN to distinguish different types. However, more and more researches pursue an efficient end-to-end lane marking discrimination model.

[106] attempts to apply the region CNN in target detection to detect lane markings and modified the framework of the anchor generator according to the slender characteristics of the lane markings. Thereby, this method realizes the anchor-based lane marking detection. However, the lane markings are detected in bounding boxes, which lacks structural continuity. To solve this problem, [66] proposes a novel anchor-based attention mechanism using global information to infer the location of lane markings. Compared to the bounding box of target detection, instance segmentation can be more accurate to the edge of the object, and differently from ordinary semantic segmentation, it can label different individuals of the same object in the images. [77] directly uses instance segmentation to recognize the type of lane marking. However, this direct segmentation of different types of lane markings is easily affected by factors such as occlusion and incomplete.

To solve this problem, [49] proposes the line prediction (LP) layer to effectively distinguish lane markings. In LineNet's LP layer (Fig. 6(a)), six detection tasks, mask, position, direction, confidence, distance and type, are separated into six branches at the end of CNN. The LP layer can also comprehensively analyze the structure information of lane markings. It is expected that each task can promote the other. However, due to the high degree of network overlap between different tasks, the different tasks will also inevitably restrict each other.

Unlike [49], multiple tasks in [44] are set in different branches belonging to different decoders to learn the structure of lane markings from many perspectives, as shown in Fig. 6(b). Except for the ordinary segmentation task used
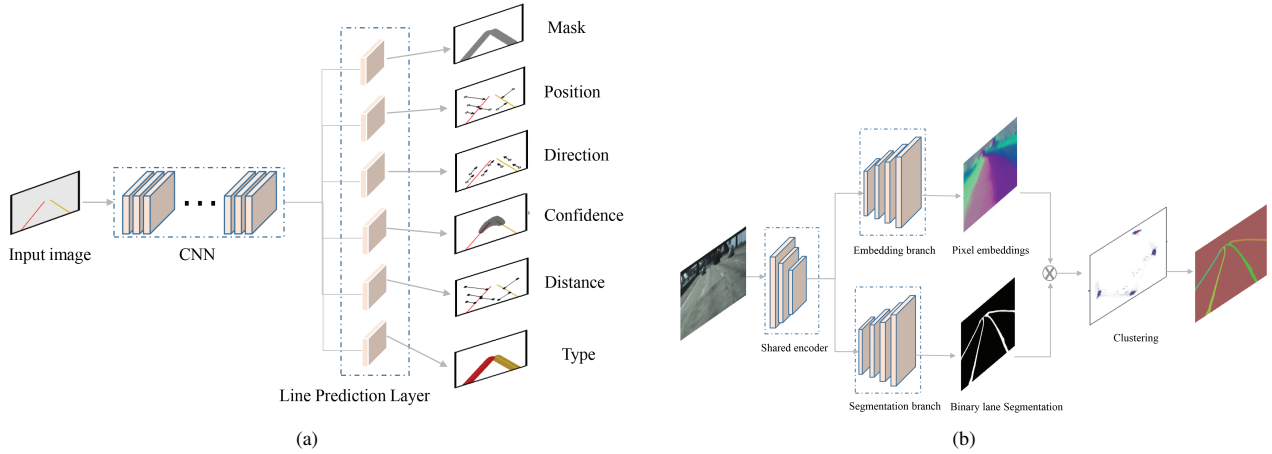
Fig. 6. Lane Marking Classification. (a) The overview of LineNet [49]. The LP layer is located at the last layer of the network and contains six branches for different tasks. (b) The overview of LaneNet in [44], where the lane marking detector is with a two-branch decoder.
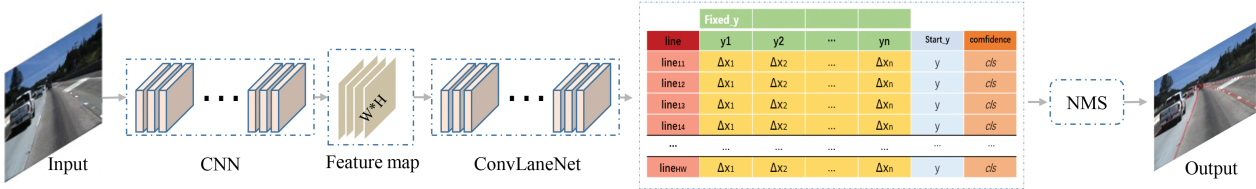

Fig. 7. The overview of PointLaneNet [64]. Two cascaded CNN networks are used for lane marking detection and classification, respectively.

to generate a binary map of lane markings, the embedding task maps the pixels of the same lane markings close and those of different lane markings far away. Benefited from the combination of those two tasks in the decoder, the problems of lane marking changes can be solved. The loss function of the segmentation branch is a standard cross-entropy loss function; the embedding branch loss function consists of two parts, as shown in Table IV (4).

Similar to [44], [59] also adopts the idea of branched multi-tasks for lane marking detection. A horizon estimation branch and a line classification branch are added after the encoder to detect the range and presence confidence of lane markings. In this way, it can also solve the problem of the change in the number of lane markings. [59] also proposes a new lane marking fitting scheme. Previous lane marking detection methods usually fit the line based on the segmentation results. Nevertheless, the lane marking segmentation could lead to unreliable detection as it is a prediction. Therefore, a differentiable least-squares fitting module is designed to directly regress lane marking parameters. A geometric loss function is proposed to achieve the end-to-end training of the network: instead of measuring the squared distance between predicted parameters and real parameters, it minimizes the distance between the predicted lane ($l_{\hat{y}}(x)$) and real curve ($l_y(x)$) to make the fitting result more realistic, as shown in Table IV (1). Experiments in [107] show that weighted least-squares linear regression on network segmentation results can fix broken lane marking edges and improve detection performance.

As multi-task learning requires post-processing to deal with different prediction results from different branches, [60] replaces multi-task branches with two cascaded CNNs to achieve instance segmentation and lane marking classification.

The methods based on pixel-by-pixel semantic segmentation have a common flaw: lane markings are only a small proportion of the whole image and a slight loss in segmentation can significantly degrade the detection performance. The regression network can avoid this flaw by directly predicting the lane marking coordinates and separating different classes of lane markings.

The coordinate network proposed by [54] can directly produce classified lane marking points. The MAE loss function is implemented to characterize the distance between predicted coordinates (xp, yp) and ground truth (xg, yg) (Table IV (2)). PointLaneNet [64] also treats lane marking detection as a regression task. Compared with [54], PointLaneNet considers more dense points including starting point, ending point (the bottom of images by default), lane marking center point and lane width (a known value according to laws and regulations) to infer disappeared lane marking features during multiple downsampling. The structure of PointLaneNet is shown in Fig. 7. PointLaneNet consists of a feature extraction network and a ConvLaneNet to conduct lane marking representation. To get more detailed detection results, the low-resolution image is divided into W×H grids to make each grid passed by one lane marking at most. Then feature maps of those girds are entered into the ConvLaneNet. The offset of lane marking

points from center points of grids are predicted together with the starting position (in the vertical direction) and lane marking presence confidence. Non-maximum suppression is finally used to suppress redundant multiple predictions for each lane marking.

It can be seen from the above discussion that the regression-based lane marking detection is easy to classify and the network is easy to converge, while the segmentation-based methods can respond flexibly to the lane marking changes. [42] combines these two methods by using the multi-task network structure. There are four tasks to exploit the structural information of lane markings in which the binary segmentation task and lane pixel embedding task are similar to those of [44]. The drivable area detection task predicts the segmentation of lanes to further constrain lane markings. Finally, there is a lane point regression task to further explore the structural and contextual information cooperating with binary segmentation results.

C. Deep architecture focusing on efficient calculation

As a task that requires real-time performance, lane marking detection also needs efficient detection speed while pursuing accuracy. In this section, we summarize some novel frameworks used in lane marking detection to improve the detection speed and finally make a comparative analysis of the previously mentioned algorithm in speed performance.

Deep networks for semantic segmentation mostly require a lot of floating-point operations. Long-running time hinders their real-time applicability. VGG tries to improve the accuracy by deepening the network. However, excessively deepening the network will make the convergence slow, and there will be over-fitting problems on small datasets. Residual network (ResNet) [108] applies residual representation to the construction of CNN models. It learns the residual representation between input and output by using short skip connections, which is much more efficient than direct learning the input-output mapping. [67] implements a small-sized anchor-based lane marking detection algorithm on ResNet, and its real-time performance can reach 175 fps. Based on ResNet, Efficient Neural Network (E-Net) [95] is designed for short processing times and has a good balance between accuracy and speed. It consists of an initial block and different bottleneck blocks.

To reduce the amount of computation, E-Net tries to use pooling and convolution parallel. Each $n \times n$ convolution is decomposed into a convolutional cascade of $n \times 1$ and $1 \times n$. This operation is called factorized convolution which can save a lot of parameters. PReLU, dilated convolutions and regularization are also applied to improve computational efficiency. E-Net is thus often used as a backbone in lane marking detection networks focusing on computational efficiency, such as [42]–[44]. Inspired by E-Net, [77] proposes Global Convolution Network which divides normal convolution into two one-dimensional convolution kernels to maximize the size of the convolution kernel without increasing the computational burden.

Unlike E-Net, which puts the downsampling on the front of the network to compress the image into a highly efficient

representation, Efficient Dense modules with Asymmetric convolution (EDANet) [96] set a 1×1 convolution in front of each EDA module to reduce the number of channels in the input map, and combine dense connectivity to increase the feature map connection between different modules. In this way, EDANet achieves a balance between accuracy and speed, which is used in [97] to achieve efficient lane marking detection. Efficient residual factorized ConvNet (ERFNet) [109] is also a network dedicated to real-time segmentation tasks. Compared with E-Net, the ERFNet further accelerates the computational efficiency. It is also based on an encoder-decoder architecture consisting of a downsampling module, a non-bottleneck-1D module and a deconvolution module.

The biggest contribution of ERFNet is the combination of accurate non-bottleneck residual connection and computationally efficient factorized convolution. In the downsampling block, the strategy of parallel pooling and convolution is also adopted. [58]–[60] use ERFNet as a backbone to improve the efficiency of lane marking detection.

The deep networks are evolving toward high efficiency and high accuracy. In addition to those mentioned in this section, there are other well-developed network structures, such as PSPNet [101] and RefineNet [110], which can also be applied to lane marking detection to balance detection accuracy and efficiency. For more comparison of real-time segmentation network performance, readers can refer to [111].

The time to test the lane markings by various methods with their experimental settings is ordered in Table V.

TABLE V
COMPARISON OF ALGORITHM COMPLEXITY. ORDERED BY FRAME PER SECOND (FPS): THE NUMBER OF IMAGES THAT CAN BE PROCESSED IN ONE SECOND.

	[76]	[67]	[66]	[47]	[61]	[43]	[59]	[64]	[60]	[44]	[63]	[58]	[38]	[98]
Parameter(M)	2.49	-	22.13	-	25.5	0.85	2.49	5.33	2.67	15.98	-	9.26	20.72	53.70
FPS	1724.1	175.4	171	98.0	90.31	74.6	71.5	71	58.93	52.6	47	-	27.58	23.81	2
Processor	GPU	GPU	GPU	GPU	-	-	GPU	GPU	GPU	GPU	GPU	GPU	GPU	GPU
#GPUs	2	1	1	1	-	-	1	-	1	1	-	1	1	1

Due to the combination of LSTM, [76] achieves a strong network inference ability by employing a simple codec, and it is the fastest when the feature from previous frames are saved and reused (although with two GPUs, its processing speed is still very competitive if converted to a single GPU). Anchor-based methods greatly improve the speed of lane marking detection since they avoid pixel-by-pixel classification in segmentation. [67] achieves faster detection than [66] by reducing the anchor size. [66] trains an efficient detection network using knowledge distillation. Both [61] and [38] are lane marking detection algorithms focusing on contexture information, however, [38] spends apparently more runtime on the proposed convolution method. [42] and [44] are similar methods that finally obtain the classified detection results through multitasking and use the E-Net as structure backbones. Since [42] performs the regression task and the segmentation task at the same time, it achieved a faster test speed eliminating the lane marking fitting step. Both [59] and [60] use ERFNet as skeleton network, but [60] has lower FPS as two cascaded CNNs are included. [98] has excellent detection performance but its network consists of basic convolution layers without paying attention to improving network computing efficiency.

Generally, lightweight networks with a small number of parameters, such as [42], [59], [76], have much less difficult to deploy.

In Section IV, we analyzed the advanced methods of lane marking detection based on the development of deep architectures. The end-to-end networks based solely on visual images are the current mainstream algorithm of lane marking detection. Also, some researches use neural networks as an aid for manual lane marking feature extraction, such as [57] and [78]. All algorithms mentioned above conduct 2D lane marking detection. We note that some algorithms have carried out related research with the aim of 3D lane marking positioning, for instance, [79] and 3D-LaneNet [80].

V. Data processing and performance evaluation

In this section, we summarize data pre-processing and post-processing techniques (Section V-A) and performance evaluation metrics (Section V-B) used by many lane marking detection methods, as well as the performance comparison (Section V-C) of these methods.

A. Effective data processing

To improve the detection results of lane markings, many researches have applied some extra but effective data pre-processing and post-processing.

Common pre-processing methods are color space change [46] [102], filter [57], data augmentation [64], [67], etc. Due to the perspective principle, the parallel lane markings intersect at a point in the distance. Inverse perspective mapping (IPM) is the inverse process of perspective mapping. It can use the positional information, such as the angle and height of the camera, to establish a 3D coordinate system, eliminate the perspective effect, and obtain the BEV of the scene. After IPM, lane markings are converted into parallel lines in BEV, which is more convenient for detection and curve fitting. IPM is used in [44], [45], [75], [79], [80], [102].

Elimination of redundant points and fitting into lines are often required as post-processing in lane marking detection. [45] and [75] employ non-maximum suppression to reduce redundancy and make the prediction more accurate. There are many ways to fit the lane markings. Second-order or third-order polynomials are often used as models for lane line fitting [48], [75], [102]; RANSAC is widely used for straight line fitting and plane fitting [78]; the least squares fit is applied in [44] and [59].

B. Evaluation metrics

From the network prediction, each entry of the confusion matrix can be obtained, including true positive (TP), true negative (TN), false positive (FP) and false negative (FN), which are the basis for common evaluation metrics.

Accuracy ($\frac{TP+TN}{TP+TN+FP+FN}$), Recall ($\frac{TP}{TP+FN}$) and Precision ($\frac{TP}{TP+FP}$) are simple and intuitive statistics which are often used to evaluate the detection performance. Recall and Precision are interdependent. When all objects are judged as positive samples, FN is 0, and the Recall reaches its maximum

of 1. However, the FP value is large at the same time, which will cause the Precision value to be very low. So it is not accurate to measure lane marking detection with Recall or Precision alone. The F1 score takes into account both Precision and Recall and is an indicator used to measure the accuracy of the two-category model in statistics:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \tag{5}$$

To solve the single value limitation of Precision, Recall and F1 score, the mean average precision (mAP) was proposed. With different number of samples are selected from the entire sample range, different values of Recall and Precision can be obtained to draw the Precision-Recall curve. AP is the area between the precision-recall curve and the coordinate axis; mAP is the average of APs in different categories.

The confidence of each prediction target should be considered when calculating the mAP. To decide prediction confidence, the intersection over union (IoU), which calculates the overlap between detection results and ground truth, is usually employed:

$$IoU = \frac{TP}{TP + FN + FP}. \tag{6}$$

For multi-category lane marking segmentation, mIoU is used instead of IoU as a metric:

$$mIoU = \frac{1}{C} \sum_{t=1}^{C} \frac{TP(t)}{TP(t) + FN(t) + FP(t)}, \tag{7}$$

where C is the number of classes.

In regression lane marking detection algorithms, the performance of detection is evaluated by the distance between the predicted value and the true value, including MSE (used in [77]), MAE (used in [77]), and mean prediction error (MPE, used in [54]),

$$MPE = \frac{1}{N} \sum_{t=1}^{N} \sqrt{(\widehat{x}_t - x_t)^2 + (\widehat{y}_t - y_t)^2}, \tag{8}$$

where $(\widehat{x}_t, \widehat{y}_t)$ are predicted coordinate of the ground truth coordinate (x_t, y_t) for N sampled points on the lane markings.

TABLE VI
REPORTED USE OF EVALUATION METRICS. *only Acc*: ONLY ACCURACY USED; *-Acc*: ONLY FN AND FP USED; *+Precision*: PRECISION ALSO USED.

Metrics	Lane marking detection methods
Accuracy, FP, FN	[53]; [42]; [43]; [61]; [60]; [64]; [76]; [66]; [65]; [50]; [52] (only Acc): [45]; [46]; [48]; [59]; [67] (-Acc): [75]; [78] (+Precision): [44]; [56]; [98]
IoU, mIoU	[38]; [49]; [43]; [58]; [47]; [48]; [54]; [61]; [51] [66]; [65]; [67]; [97]; [68]; [99]; [100]; [52]
F1 score	[38]; [49]; [42]; [43]; [47]; [56]; [51]; [62]; [64] [76]; [65]; [66]; [67]; [68]; [98]; [99]; [50]; [52]
MSE, MAE	[77]; [79]
MPE	[57]; [54]; [102]
mAP	[100]

The reported use of the evaluation metrics is summarized in Table VI. Algorithms using the Tusimple dataset would use accuracy, FP and FN, as they are the metrics used in the Tusimple lane marking detection competition.

C. Performance comparison

Tusimple and CULane are two datasets most commonly used to evaluate the algorithms mentioned in Section IV.

TABLE VII
RESULTS ON THE TUSIMPLE DATASET. DA: DATA AUGMENTATION. RES34: FRAMEWORK BASED OM RESNET-34

	F1 score	Accuracy	FP	FN	Extra training data	Supervision
[98]	**97.67**	**97.82**	**0.0048**	0.0289	No	semi-supervision
[66](Res34)	96.77	95.63	0.0353	0.0292	No	supervision
[52]	93.44	96.91	0.0263	0.0252	Yes(DA)	supervision
[65](Res34)	92.28	96.22	0.0308	0.0376	No	supervision
[56]	91.13	95.25	-	-	No	supervision
[53]	90.56	96.39	0.0412	0.0336	No	semi-supervision
[64]	89.61	96.34	0.0467	0.0518	Yes(DA)	supervision
[43]	87.56	96.64	0.0602	0.0205	No	supervision
[38]	87.27	96.53	0.0617	**0.0180**	Yes	supervision
[42]	85.52	96.29	0.0722	0.0218	No	supervision
[44]	84.71	96.40	0.0780	0.0244	No	supervision
[61]	84.39	95.20	0.0760	0.0450	Yes	semi-supervision
[60]	78.55	95.24	0.1197	0.0620	No	supervision
[50]	66.06	96.51	0.2393	0.0316	Yes(DA)	supervision
[67](Res34)	-	96.06	-	-	Yes(DA)	supervision
[59]	-	95.80	-	-	No	supervision

The performance comparison on Tusimple is shown in Table VII, ordered by the F1 scores. In lane marking segmentation algorithms, [98] and [53] use GAN to learn the distribution of lane marking, making the algorithm have a smaller FP and achieve the excellent F1 score. For some algorithms based on multiple network frameworks, the best performance has been selected for display. The anchor-based method [66] achieves outstanding detection performance (better than a similar method [60]) by using an attention mechanism to integrate context information. [65] improves the detection accuracy with the help of row classification. The novel convolution module SCNN helps [38] obtain a very low FN, but it also misjudges more points as lane markings, resulting in higher FP. In lane marking regression algorithms, [64] regressed lane marking position on segmentation feature map and partly benefited from data augmentation, performing better than many lane marking segmentation methods.

TABLE VIII
F1 SCORE ON THE CULANE DATASET. ALL ALGORITHMS ARE SUPERVISED LEARNING. CR: CROSSROAD; DA: DATA AUGMENTATION. RES34: FRAMEWORK BASED ON RESNET-34. ERFNET: FRAMEWORK BASED ON ERFNET.

	Total	Normal	Crowd	Night	No line	Shadow	Arrow	Dazzle light	Curve	CR (FP)	Extra data (training)
[66](Res34)	**76.7**	**92.1**	**75.0**	**70.7**	**49.4**	**78.2**	**88.4**	66.5	67.7	**1330**	No
[51]	74.8	90.7	72.3	68.9	49.4	70.1	85.8	**67.7**	68.4	1746	No
[52]	74.6	92.0	72.9	70.2	48.2	70.6	87.4	67.5	**72.4**	2849	Yes(AD)
[99]	74.2	91.9	72.3	69.4	46.8	74.0	87.4	67.1	66.4	2292	No
[68]	74.2	90.5	71.7	68.4	43.2	72.9	85.0	62.0	50.7	-	No
[65](ERFNet)	74.0	91.0	73.1	67.9	46.6	74.1	85.8	64.5	71.9	2022	No
[49]	73.1	-	-	-	-	-	-	-	-	-	Yes
[47]	72.4	-	-	-	-	-	-	-	-	-	No
[67](Res34)	72.3	90.7	70.2	66.7	44.4	69.3	85.7	59.5	69.5	2037	Yes(DA)
[38]	71.6	90.6	69.7	66.1	43.4	66.9	84.1	58.5	64.4	1990	Yes
[43]	70.8	-	-	-	-	-	-	-	-	-	No
[42]	68.8	88.4	67.0	64.1	42.9	63.4	81.9	57.4	62.6	2768	No
[50]	-	90.2	69.7	67.3	44.7	68.5	84.8	59.7	69.6	1933	Yes(DA)
[64]	-	90.1	-	-	-	-	-	-	-	-	Yes(DA)
[61]	-	85.9	63.6	57.8	40.6	59.9	79.4	57.0	65.2	7013	Yes

CULane consists of multiple challenging traffic scenes as listed in Table VIII. The anchor-based attention mechanism helps [66] achieve the best scores in most traffic scenes, except for Dazzle light and Curve, where NAS-based [51] with long-ranged coherent and detailed curve trajectory performs better. Due to the deconstruction of lane markings by channel attention, [52] obtains excellent curve detection performance. Regarding the impact of novel convolution module on the detection performance, SpinNet [68] is better than SCNN [38],

because it incorporates more directional features. Compared with self-attention distillation [43], the teacher network has a great contribution to improving the performance of the student network, which makes [47] achieve better detection performance. [49] attains better performance via multi-task and multi-resolution prediction. In short, more complex scenarios require a richer representation of features.

VI. OUTLOOK

Intelligent driving vehicles take the safety risk due to the uncertainty in real-world traffic scenes. As an important part of intelligent driving risk assessment, accurate lane marking detection can help segment and understand traffic scenes efficiently, which is beneficial to subsequent path planning and motion planning [21]. Existing approaches to deep learning-based lane marking detection exploit the ability of segmentation and regression networks to extract lane markings in images. They aim to learn discriminative features of lane markings and infer lane marking positions in a variety of challenging scenarios. However, their performance is still only satisfactory in simple scenes. In the case of occlusion or missing lane markings, there remain some challenges and opportunities for improvement.

First, many current researches are dedicated to designing a universal lane marking detection algorithm. Such an algorithm, however, often performs poorly under some special traffic scenes (e.g., extreme weather conditions). Solutions to specific challenging scenarios are a practically useful direction meriting further investigation.

Secondly, lane markings have certain distributional laws, which have not been fully exploited yet, e.g., the mutual reasoning between lane markings to address the situations of occlusions and missing lane markings. Hence, the contextual structure of pixels of lane markings can be embedded in the design of networks to enhance lane marking detection.

Thirdly, objects other than lane markings in global scenes, such as cars and fences, can assist the detection of lane markings. That is, the network can be designed to not only focus on lane markings but also make more use of advanced semantic information in the entire scene to guide detection.

Fourthly, lane marking detection has a high requirement for both accurate detection and real-time performance and thus an important concern about the trade-off between computing power and algorithm complexity. Hence, we need to consider whether the complexity of deep learning is acceptable to embedded systems. To address this challenge, we need to consider integrating relevant techniques, such as video summarization [112], energy-friendly edge intelligence-assisted algorithm [113] and economic hardware selection [114].

Finally, existing deep learning-based lane marking detection algorithms extremely rely on datasets. However, existing datasets will be slowly phased out with the increasing complexity of traffic scenarios. To weaken the dependence of datasets, improving the autonomous learning ability of the network is an effective way of meriting investigation, e.g., exploring self-supervised learning in lane marking detection.

In summary, the lane marking detection research has moved from solving problems in generally simple scenes to the stage

where it needs to perform in complex scenes. While focusing on improving detection performance, we also need to pay attention to the practicality of algorithms, e.g., to reduce the complexity of algorithms and promote real-time performance.

VII. CONCLUSION

In this paper, we review the state-of-art deep networks for lane marking detection. We summarize the available datasets, objective functions, evaluation criteria and data processing techniques, and elaborate novel network designs for each of these methods in more detail including comparing their performances and running times. We also discuss the current status and challenges of lane marking detection and put forward some suggestions. To help readers quickly understand various algorithms, some characteristics of representative methods mentioned in this survey are summarized in Table IX.

REFERENCES

[1] R. Bishop, A survey of intelligent vehicle applications worldwide, in: Proceedings of the IEEE Intelligent Vehicles Symposium, IEEE, 2000, pp. 25–30.

[2] E. Q. Wu, M. Zhou, D. Hu, L. Zhu, Z. Tang, X.-Y. Qiu, P.-Y. Deng, L.-M. Zhu, H. Ren, Self-paced dynamic infinite mixture model for fatigue evaluation of pilots' brains, IEEE Transactions on Cybernetics (2020).

[3] E. Q. Wu, L.-M. Zhu, W.-M. Zhang, P.-Y. Deng, B. Jia, S.-D. Chen, H. Ren, G.-R. Zhou, Novel nonlinear approach for real-time fatigue eeg data: An infinitely warped model of weighted permutation entropy, IEEE Transactions on Intelligent Transportation Systems (2019).

[4] M. R. Minar, J. Naher, Recent advances in deep learning: an overview, arXiv preprint arXiv:1807.08169 (2018).

[5] K. Arulkumaran, M. P. Deisenroth, M. Brundage, A. A. Bharath, Deep reinforcement learning: A brief survey, IEEE Signal Processing Magazine 34 (6) (2017) 26–38.

[6] F.-Y. Wang, N.-N. Zheng, D. Cao, C. M. Martinez, L. Li, T. Liu, Parallel driving in CPSS: A unified approach for transport automation and vehicle intelligence, IEEE/CAA Journal of Automatica Sinica 4 (4) (2017) 577–587.

[7] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.

[8] G. E. Hinton, R. R. Salakhutdinov, Reducing the dimensionality of data with neural networks, science 313 (5786) (2006) 504–507.

[9] E. Q. Wu, G.-R. Zhou, L.-M. Zhu, C.-F. Wei, H. Ren, R. S. Sheng, Rotated sphere Haar wavelet and deep contractive auto-encoder network with fuzzy Gaussian SVM for pilot's pupil center detection, IEEE transactions on cybernetics (2019).

[10] G. E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural computation 18 (7) (2006) 1527–1554.

[11] G. Wang, J. Qiao, J. Bi, W. Li, M. Zhou, TL-GDBN: Growing deep belief network with transfer learning, IEEE Transactions on Automation Science and Engineering 16 (2) (2018) 874–885.

[12] E. Q. Wu, D. Hu, P.-Y. Deng, Z. Tang, Y. Cao, W.-M. Zhang, L.-M. Zhu, H. Ren, Nonparametric bayesian prior inducing deep network for automatic detection of cognitive status, IEEE Transactions on Cybernetics (2020).

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in neural information processing systems, 2014, pp. 2672–2680.

[14] Y. Lv, Y. Chen, L. Li, F.-Y. Wang, Generative adversarial networks for parallel transportation systems, IEEE Intelligent Transportation Systems Magazine 10 (3) (2018) 4–10.

[15] J. Schmidhuber, Deep learning in neural networks: An overview, Neural networks 61 (2015) 85–117.

[16] Z. Cao, X. Xu, B. Hu, M. Zhou, Rapid detection of blind roads and crosswalks by using a lightweight semantic segmentation network, IEEE Transactions on Intelligent Transportation Systems (2020).

[17] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, F.-Y. Wang, Driver activity recognition for intelligent vehicles: A deep learning approach, IEEE Transactions on Vehicular Technology 68 (6) (2019) 5379–5390.

[18] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, S. Khudanpur, Recurrent neural network based language model, in: Eleventh annual conference of the international speech communication association, 2010.

[19] Y. Chen, Y. Lv, X. Wang, L. Li, F.-Y. Wang, Detecting traffic information from social media texts with deep learning approaches, IEEE Transactions on Intelligent Transportation Systems 20 (8) (2018) 3049–3058.

[20] H. Wang, D.-Y. Yeung, Towards Bayesian deep learning: A survey, arXiv preprint arXiv:1604.01662 (2016).

[21] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, V. H. C. de Albuquerque, Deep learning for safe autonomous driving: Current challenges and future directions, IEEE Transactions on Intelligent Transportation Systems (2020).

[22] T. Rateke, K. A. Justen, V. F. Chiarella, A. C. Sobieranski, E. Comunello, A. V. Wangenheim, Passive vision region-based road detection: A literature review, ACM Computing Surveys 52 (2) (2019) 31.

[23] J. Pan, H. Sun, K. Xu, Y. Jiang, X. Xiao, J. Hu, J. Miao, Lane attention: Predicting vehicles' moving trajectories by learning their attention over lanes, arXiv preprint arXiv:1909.13377 (2019).

[24] V. Rin, C. Nuthong, Front moving vehicle detection and tracking with Kalman filter, in: International Conference on Computer and Communication Systems, IEEE, 2019, pp. 304–310.

[25] M. Cattaruzza, Design and simulation of autonomous driving algorithms, Ph.D. thesis, Politecnico di Torino (2019).

[26] L.-H. Juang, J.-S. Zhang, Robust visual line-following navigation system for humanoid robots, Artificial Intelligence Review (2018) 1–18.

[27] A. Mancini, E. Frontoni, P. Zingaretti, Mechatronic system to help visually impaired users during walking and running, IEEE Transactions on Intelligent Transportation Systems 19 (2) (2018) 649–660.

[28] A. B. Hillel, R. Lerner, D. Levi, G. Raz, Recent progress in road and lane detection: a survey, Machine vision and applications 25 (3) (2014) 727–745.

[29] S. Yenikaya, G. Yenikaya, E. Düven, Keeping the vehicle on the road: A survey on on-road lane detection systems, ACM Computing Surveys 46 (1) (2013) 2.

[30] W. Yang, X. Yang, Y. Wei, X. Cheng, Study of the recognition and tracking methods for lane lines based on image edge detections, in: International Symposium on Communication Engineering & Computer Science, Atlantis Press, 2018, pp. 179–185.

[31] S. Yang, J. Wu, Y. Shan, Y. Yu, S. Zhang, A novel vision-based framework for real-time lane detection and tracking, Tech. rep., SAE Technical Paper (2019).

[32] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, Dynamic integration and online evaluation of vision-based lane detection algorithms, IET Intelligent Transport Systems 13 (1) (2018) 55–62.

[33] R. E. Haas, S. Bhattacharjee, D. P. Möller, Advanced driver assistance systems, in: Smart Technologies, Springer, 2020, pp. 345–371.

[34] M. A. Da Conceicao, et al., Line lane tracker computation using robust feature extraction for autonomous vehicle, Journal of the Gujarat Research Society 21 (4) (2019) 199–205.

[35] F. Zheng, S. Luo, K. Song, C.-W. Yan, M.-C. Wang, Improved lane line detection algorithm based on Hough transform, Pattern Recognition and Image Analysis 28 (2) (2018) 254–260.

[36] D. Zeng, G. Zeng, P. O. Kodom, Research on recognition technology of vehicle rolling line violation in highway based on visual UAV, in: International Conference on Robotics, Control and Automation, ACM, 2019, pp. 198–204.

[37] J. Cao, C. Song, S. Song, F. Xiao, S. Peng, Lane detection algorithm for intelligent vehicles in complex road conditions and dynamic environments, Sensors 19 (14) (2019) 3166.

[38] X. Pan, J. Shi, P. Luo, X. Wang, X. Tang, Spatial as deep: Spatial CNN for traffic scene understanding, in: AAAI Conference on Artificial Intelligence, 2018, pp. 7276–7283.

[39] Y. Xing, C. Lv, L. Chen, H. Wang, H. Wang, D. Cao, E. Velenis, F.-Y. Wang, Advances in vision-based lane detection: algorithms, integration, assessment, and perspectives on ACP-based parallel vision, IEEE/CAA Journal of Automatica Sinica 5 (3) (2018) 645–661.

[40] S. P. Narote, P. N. Bhujbal, A. S. Narote, D. M. Dhane, A review of recent advances in lane detection and departure warning system, Pattern Recognition 73 (2018) 216–234.

[41] J. Tang, S. Li, P. Liu, A review of lane detection methods based on deep learning, Pattern Recognition (2020) 107623.

[42] Y. Hou, Agnostic lane detection, arXiv preprint arXiv:1905.03704 (2019).

TABLE IX

SUMMARY OF REPRESENTATIVE METHODS. CLASSIFICATION: LANE MARKING CLASSIFICATION. THE POST-PROCESSING HERE DOES NOT INCLUDE THE CURVE FITTING STEPS REQUIRED FOR THE CULANE DATASET.

Method	Input	Output	Classification	Supervision	Dataset	Pre-process	Post-process	Key Words
LineNet [49]	single frame	segmentation map	✓	supervision	CULane; specific	×	✓	Multiscale; Multitasking
SAD [43]	single frame	segmentation map	×	supervision	BDD100K; CULane; Tusimple	×	×	Self Attention Distillation
Affinity Distillation [47]	single frame	segmentation map	×	supervision	ApolloSpape; CULane; LLAMAS	×	×	Teacher-student Distillation
Attention DNN [62]	single frame	segmentation map	✓	supervision	CULane; Tusimple	×	×	Self Attention; Channel Attention
SALMNet [52]	single frame	segmentation map	×	supervision	BDD100K; CULane; Tusimple	×	×	Semantic-guided Channel Attention Pyramid Deformable Convolution
LMD [48]	single frame	segmentation map	×	supervision	Camvid	×	✓	Dilated Convolution
Multi-class ECN [97]	single frame	segmentation map	✓	supervision	specific	×	×	Modified EDANet
SCNN [38]	single frame	segmentation map	✓	supervision	Cityscape; CULane; Tusimple	×	×	Spatial CNN
SpinNet [68]	single frame	segmentation map	✓	supervision	CULane	×	×	Spinning Convolution
GCN [77]	single frame	segmentation map	✓	supervision	specific	×	×	Graph Convolutional Network
EL-GAN [53]	single frame	segmentation map	×	semi-supervision	Tusimple	×	×	GAN
Ripple-GAN [98]	single frame	segmentation map	×	semi-supervision	Tusimple	✓	×	GAN
VPGNet [55]	single frame	segmentation map	✓	supervision	Caltech; VPGNet	×	✓	Vanish Point
ERFNet-VP [99]	single frame	segmentation map	✓	supervision	CULane	×	×	Heatmap-based Vanish Point
Multitask attention network [50]	single frame	segmentation map	✓	supervision	Caltech; CULane; KITTY Road; Tusimple	×	×	Vanishing Line
Lane departure detection [100]	single frame	segmentation map	×	supervision	specific	×	✓	Indirect Detection
Free space detection [58]	single frame	segmentation map	✓	supervision	BDD100K	×	✓	Indirect Detection
FastDraw [61]	single frame	segmentation map	×	semi-supervision	CULane; Tusimple	×	×	Autoregression
CurveLane-NAS [51]	single frame	segmentation map	×	supervision	CULane; CurveLane; Tusimple	×	✓	NAS-based
E2E-LMD [65]	single frame	key points	✓	supervision	CULane; Tusimple	×	×	Row-wise Classification
LaneNet [75]	single frame	segmentation map	✓	supervision	specific	✓	×	LSTM-based
ConvLSTM [76]	sequential frames	segmentation map	×	supervision	specific	×	×	LSTM-based
Spatio-Temporal Network [56]	sequential frames	segmentation map	×	supervision	LLAMAS; Tusimple	×	×	LSTM-based
Optical Lane Detection [63]	sequential frames	segmentation map	×	supervision	specific; Tusimple	✓	✓	Optical Flow Estimation
Lane marking detection [106]	single frame	proposal anchor	×	supervision	specific; KITTY Road	×	×	Anchor-based
LaneATT [66]	single frame	key points	✓	supervision	CULane; Tusimple	×	✓	Anchor-based Attention
LaneNet [44]	single frame	segmentation map	✓	supervision	Tusimple	✓	✓	Multitask
End-to-end detection [59]	single frame	lane parameters	✓	supervision	Tusimple	×	✓	Differentiable Least-squares Fitting
Cascaded CNNs [60]	single frame	segmentation map	✓	supervision	Tusimple	×	×	Cascaded CNN
Reliable multilane detection [54]	single frame	key points	✓	supervision	Tusimple	×	×	Regression
PointLaneNet [64]	single frame	key points	✓	supervision	CULane; Tusimple	✓	✓	Regression
Agnostic lane detection [42]	single frame	segmentation map	✓	supervision	BDD100K; CULane; Tusimple	×	✓	Multitask
Ultra Fast [67]	single frame	key points	✓	supervision	CULane; Tusimple	✓	×	Row-wise Classification

[43] Y. Hou, Z. Ma, C. Liu, C. C. Loy, Learning lightweight lane detection CNNs by self attention distillation, in: IEEE International Conference on Computer Vision, 2019, pp. 1013–1021.

[44] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, L. Van Gool, Towards end-to-end lane detection: an instance segmentation approach, in: IEEE Intelligent Vehicles Symposium, IEEE, 2018, pp. 286–291.

[45] W. Song, Y. Yang, M. Fu, Y. Li, M. Wang, Lane detection and classification for forward collision warning system based on stereo vision, IEEE Sensors Journal 18 (12) (2018) 5151–5163.

[46] R. Fan, N. Dahnoun, Real-time stereo vision-based lane detection system, Measurement Science and Technology 29 (7) (2018) 074005.

[47] Y. Hou, Z. Ma, C. Liu, T.-W. Hui, C. C. Loy, Inter-region affinity distillation for road marking segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12486–12495.

[48] P.-R. Chen, S.-Y. Lo, H.-M. Hang, S.-W. Chan, J.-J. Lin, Efficient road lane marking detection with deep learning, in: IEEE International Conference on Digital Signal Processing, IEEE, 2018, pp. 1–5.

[49] D. Liang, Y. Guo, S. Zhang, S.-H. Zhang, P. Hall, M. Zhang, S. Hu, LineNet: a zoomable CNN for crowdsourced high definition maps modeling in urban environments, arXiv preprint arXiv:1807.05696 (2018).

[50] Q. Wang, T. Han, Z. Qin, J. Gao, X. Li, Multitask attention network for lane detection and fitting, IEEE transactions on neural networks and learning systems (2020).

[51] H. Xu, S. Wang, X. Cai, W. Zhang, X. Liang, Z. Li, CurveLane-NAS: Unifying lane-sensitive architecture search and adaptive point blending, in: European Conference on Computer Vision, 2020, pp. 689–704.

[52] X. Xu, T. Yu, X. Hu, W. W. Ng, P.-A. Heng, Salmnet: A structure-aware lane marking detection network, IEEE Transactions on Intelligent Transportation Systems (2020).

[53] M. Ghafoorian, C. Nugteren, N. Baka, O. Booij, M. Hofmann, EL-GAN: embedding loss driven generative adversarial networks for lane detection, in: European Conference on Computer Vision, Springer, 2018, pp. 256–272.

[54] S. Chougule, N. Koznek, A. Ismail, G. Adam, V. Narayan, M. Schulze, Reliable multilane detection and classification by utilizing CNN as a regression network, in: European Conference on Computer Vision, Springer, 2018, pp. 740–752.

[55] S. Lee, J. Kim, J. Shin Yoon, S. Shin, O. Bailo, N. Kim, T.-H. Lee, H. Seok Hong, S.-H. Han, I. So Kweon, VPGNet: Vanishing point guided network for lane and road marking detection and recognition, in:

[56] J. Zhang, T. Deng, F. Yan, W. Liu, Lane detection model based on spatio-temporal network with double convgrus, arXiv preprint arXiv:2008.03922 (2020).

[57] T. Gupta, H. S. Sikchi, D. Charkravarty, Robust lane detection using multiple features, in: IEEE Intelligent Vehicles Symposium, IEEE, 2018, pp. 1470–1475.

[58] F. Pizzati, F. García, Enhanced free space detection in multiple lanes based on single CNN with scene identification, arXiv preprint arXiv:1905.00941 (2019).

[59] W. Van Gansbeke, B. De Brabandere, D. Neven, M. Proesmans, L. Van Gool, End-to-end lane detection through differentiable least-squares fitting, in: IEEE International Conference on Computer Vision Workshops, 2019, pp. 0–0.

[60] F. Pizzati, M. Allodi, A. Barrera, F. García, Lane detection and classification using cascaded CNNs, arXiv preprint arXiv:1907.01294 (2019).

[61] J. Philion, FastDraw: Addressing the long tail of lane detection by adapting a sequential prediction network, in: IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11582–11591.

[62] D. Xiao, X. Yang, J. Li, M. Islam, Attention deep neural network for lane marking detection, Knowledge-Based Systems (2020) 105584.

[63] S. Lu, Z. Luo, F. Gao, M. Liu, K. Chang, C. Piao, A fast and robust lane detection method based on semantic segmentation and optical flow estimation, Sensors 21 (2) (2021) 400.

[64] Z. Chen, Q. Liu, C. Lian, PointLaneNet: Efficient end-to-end CNNs for accurate real-time lane detection, in: IEEE Intelligent Vehicles Symposium, IEEE, 2019, pp. 2563–2568.

[65] S. Yoo, H. S. Lee, H. Myeong, S. Yun, H. Park, J. Cho, D. H. Kim, End-to-end lane marker detection via row-wise classification, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020, pp. 4335–4343.

[66] L. Tabelini, R. Berriel, T. M. Paixão, C. Badue, A. F. De Souza, T. Olivera-Santos, Keep your eyes on the lane: Attention-guided lane detection, arXiv preprint arXiv:2010.12035 (2020).

[67] Z. Qin, H. Wang, X. Li, Ultra fast structure-aware deep lane detection, in: European Conference on Computer Vision, 2020, pp. 276–291.

[68] R. Fan, X. Wang, Q. Hou, H. Liu, T.-J. Mu, Spinnet: Spinning convolutional network for lane boundary detection, Computational Visual Media 5 (4) (2019) 417–428.

[69] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: The KITTI dataset, International Journal of Robotics Research (2013).

[70] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, T. Darrell, BDD100K: A diverse driving video database with scalable annotation tooling, arXiv preprint arXiv:1805.04687 (2018).

[71] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The Cityscapes dataset for semantic urban scene understanding, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.

[72] G. J. Brostow, J. Fauqueur, R. Cipolla, Semantic object classes in video: A high-definition ground truth database, Pattern Recognition Letters 30 (2) (2009) 88–97.

[73] K. Behrendt, R. Soussan, Unsupervised labeled lane markers using maps, in: IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), 2019, pp. 832–839.

[74] W. Cheng, H. Luo, W. Yang, L. Yu, W. Li, Structure-aware network for lane marker extraction with dynamic vision sensor, arXiv preprint arXiv:2008.06204 (2020).

[75] Z. Wang, W. Ren, Q. Qiu, LaneNet: Real-time lane detection networks for autonomous driving, arXiv preprint arXiv:1807.01726 (2018).

[76] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, Q. Wang, Robust lane detection from continuous driving scenes using deep neural networks, arXiv preprint arXiv:1903.02193 (2019).

[77] W. Zhang, T. Mahale, End to end video segmentation for driving: Lane detection for autonomous car, arXiv preprint arXiv:1812.05914 (2018).

[78] Z. Lu, Y. Xu, X. Shan, L. Liu, X. Wang, J. Shen, A lane detection method based on a ridge detector and regional G-RANSAC, Sensors 19 (18) (2019) 4028.

[79] M. Bai, G. Mattyus, N. Homayounfar, S. Wang, S. K. Lakshmikanth, R. Urtasun, Deep multi-sensor lane detection, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2018, pp. 3102–3109.

[80] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, D. Levi, 3D-LaneNet: end-to-end 3D multiple lane detection, in: IEEE International Conference on Computer Vision, 2019, pp. 2921–2930.

[81] J. T. Barron, A general and adaptive robust loss function, in: IEEE conference on Computer Vision and Pattern Recognition, 2019, pp. 4331–4339.

[82] A. Van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al., Conditional image generation with PixelCNN decoders, in: Advances in neural information processing systems, 2016, pp. 4790–4798.

[83] I. Kobyzev, S. Prince, M. Brubaker, Normalizing flows: An introduction and review of current methods, IEEE Transactions on Pattern Analysis and Machine Intelligence (2020).

[84] D. P. Kingma, M. Welling, Auto-encoding variational Bayes, arXiv preprint arXiv:1312.6114 (2013).

[85] A. Müller, Integral probability metrics and their generating classes of functions, Advances in Applied Probability (1997) 429–443.

[86] S. Nowozin, B. Cseke, R. Tomioka, F-GAN: Training generative neural samplers using variational divergence minimization, in: Advances in neural information processing systems, 2016, pp. 271–279.

[87] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein GAN, arXiv preprint arXiv:1701.07875 (2017).

[88] J. Kim, M. Lee, Robust lane detection based on convolutional neural network and random sample consensus, in: International conference on neural information processing, Springer, 2014, pp. 454–461.

[89] D. C. Cireşan, A. Giusti, L. M. Gambardella, J. Schmidhuber, Mitosis detection in breast cancer histology images with deep neural networks, in: International conference on medical image computing and computer-assisted intervention, Springer, 2013, pp. 411–418.

[90] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, K. Zieba, End to end learning for self-driving cars, arXiv preprint arXiv:1604.07316 (2016).

[91] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: IEEE conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

[92] J. Zang, W. Zhou, G. Zhang, Z. Duan, Traffic lane detection using fully convolutional neural network, in: Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, IEEE, 2018, pp. 305–311.

[93] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 234–241.

[94] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, arXiv preprint arXiv:1511.07122 (2015).

[95] A. Paszke, A. Chaurasia, S. Kim, E. Culurciello, ENet: A deep neural network architecture for real-time semantic segmentation, arXiv preprint arXiv:1606.02147 (2016).

[96] S.-Y. Lo, H.-M. Hang, S.-W. Chan, J.-J. Lin, Efficient dense modules of asymmetric convolution for real-time semantic segmentation, in: Proceedings of the ACM Multimedia Asia, 2019, pp. 1–6.

[97] S.-Y. Lo, H.-M. Hang, S.-W. Chan, J.-J. Lin, Multi-class lane semantic segmentation using efficient convolutional networks, in: IEEE International Workshop on Multimedia Signal Processing, IEEE, 2019, pp. 1–6.

[98] Y. Zhang, Z. Lu, D. Ma, J.-H. Xue, Q. Liao, Ripple-GAN: Lane line detection with ripple lane line detection network and wasserstein GAN, IEEE Transactions on Intelligent Transportation Systems (2020) 1–11.

[99] Y.-B. Liu, M. Zeng, Q.-H. Meng, Heatmap-based vanishing point boosts lane detection, arXiv preprint arXiv:2007.15602 (2020).

[100] L. Riera, K. Ozcan, J. Merickel, M. Rizzo, S. Sarkar, A. Sharma, Driver behavior analysis using lane departure detection under challenging conditions, arXiv preprint arXiv:1906.00093 (2019).

[101] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 2881–2890.

[102] Y. Ma, V. Havyarimana, J. Bai, Z. Xiao, Vision-based lane detection and lane-marking model inference: A three-step deep learning approach, in: International Symposium on Parallel Architectures, Algorithms and Programming, IEEE, 2018, pp. 183–190.

[103] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: IEEE conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.

[104] C.-A. Brust, S. Sickert, M. Simon, E. Rodner, J. Denzler, Convolutional patch networks with spatial prior for road detection and urban scene understanding, arXiv preprint arXiv:1502.06344 (2015).

[105] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, Z. Wang, Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network, in: IEEE conference on Computer Vision and Pattern Recognition, 2016, pp. 1874–1883.

[106] Y. Tian, J. Gelernter, X. Wang, W. Chen, J. Gao, Y. Zhang, X. Li, Lane marking detection via deep convolutional neural network, Neurocomputing 280 (2018) 46–55.

[107] Z. M. Chng, J. M. H. Lew, J. A. Lee, Roneld: Robust neural network output enhancement for active lane detection, arXiv preprint arXiv:2010.09548 (2020).

[108] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[109] E. Romera, J. M. Alvarez, L. M. Bergasa, R. Arroyo, ErfNet: Efficient residual factorized ConvNet for real-time semantic segmentation, IEEE Transactions on Intelligent Transportation Systems 19 (1) (2017) 263–272.

[110] G. Lin, A. Milan, C. Shen, I. Reid, RefineNet: Multi-path refinement networks for high-resolution semantic segmentation, in: IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 1925–1934.

[111] A. Briot, P. Viswanath, S. Yogamani, Analysis of efficient CNN design techniques for semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 663–672.

[112] K. Muhammad, T. Hussain, J. Del Ser, V. Palade, V. H. C. De Albuquerque, DeepReS: A deep learning-based video summarization strategy for resource-constrained industrial surveillance scenarios, IEEE Transactions on Industrial Informatics 16 (9) (2019) 5938–5947.

[113] K. Muhammad, S. Khan, V. Palade, I. Mehmood, V. H. C. De Albuquerque, Edge intelligence-assisted smoke detection in foggy surveillance environments, IEEE Transactions on Industrial Informatics 16 (2) (2019) 1067–1075.

[114] M. Sajjad, M. Irfan, K. Muhammad, J. Del Ser, J. Sanchez-Medina, S. Andreev, W. Ding, J. W. Lee, An efficient and scalable simulation model for autonomous vehicles with economical hardware, IEEE Transactions on Intelligent Transportation Systems (2020).