



Article

Dynamic Data Augmentation Based on Imitating Real Scene for Lane Line Detection

Qingwang Wang ^{1,2} , Lu Wang ^{1,2}, Yongke Chi ^{1,2}, Tao Shen ^{1,2,*}, Jian Song ¹ , Ju Gao ³ and Shiquan Shen ⁴

¹ Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

² Yunnan Key Laboratory of Computer Technologies Application, Kunming University of Science and Technology, Kunming 650500, China

³ Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China

⁴ Faculty of Transportation Engineering, Kunming University of Science and Technology, Kunming 650500, China

* Correspondence: shentao@kust.edu.cn; Tel.: +86-0871-65916593

Abstract: With the rapid development of urban ground transportation, lane line detection is gradually becoming a major technological direction to help to realize safe vehicle navigation. However, lane line detection results may have incompleteness issues, such as blurry lane lines and disappearance of the lane lines in the distance, since the lane lines may be heavily obscured by vehicles and pedestrians on the road. In addition, low-visibility environments also pose a challenge for lane line detection. To solve the above problems, we propose a dynamic data augmentation framework based on imitating real scenes (DDA-IRS). DDA-IRS contains three data augmentation strategies that simulate different realistic scenes (i.e., shadows, dazzle, and crowded). In this way, we expand from a limited scene dataset to realistically fit multiple complex scenes. Importantly, DDA-IRS is a lightweight framework that can be integrated with a variety of training-based models without modifying the original model. We evaluate the proposed DDA-IRS on the CULane dataset, and the results show that the data-enhanced model outperforms the baseline model by 0.5% in terms of F-measure. In particular, the F-measure of the “Normal”, “Crowded”, “Shadow”, “Arrow”, and “Curve” achieve a 0.4%, 0.1%, 1.6%, 0.4%, and 1.4% improvement, respectively.

Keywords: dynamic data augmentation; imitating real scene; lane line detection; vehicle navigation; urban ground transportation



Citation: Wang, Q.; Wang, L.; Chi, Y.; Shen, T.; Song, J.; Gao, J.; Shen, S. Dynamic Data Augmentation Based on Imitating Real Scene for Lane Line Detection. *Remote Sens.* **2023**, *15*, 1212. <https://doi.org/10.3390/rs15051212>

Academic Editor: Andrzej Stateczny

Received: 10 January 2023

Revised: 17 February 2023

Accepted: 19 February 2023

Published: 22 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the development of modern urban transportation, the road network is becoming more and more complex. Relying on human memory to guide the traffic driving routes to their destinations is becoming unrealistic, so car navigation plays an increasingly important role. Conventional car navigation is based on GPS, which first locates the user's vehicle on the road network map, then plans the driving routes according to the selected destinations and guides the user to the destination through on-screen displays and voice announcements. Emerging car navigation incorporates GPS and self-driving technology, which automatically helps drivers make reasonable decisions by combining navigation information with real road scenario information, and then allows the car to take part in the navigation driving function, thereby reducing the driver's driving and understanding costs. In particular, real-time lane line detection provides the emerging navigation engine with information such as the current lane width and lane line properties that enables the car to drive automatically into the correct lane.

Lane line detection is one of the most fundamental tasks of perception for both self-driving and advanced driver assistant systems (ADAS). By identifying the lane lines of the current vehicle driving environment, navigation paths can be provided for vehicle route

planning, allowing vehicles to move safely and controllably in a standard road model. Given the advantages of visual sensors and the feasibility of image processing technology, the use of visual navigation has become an important development direction in intelligent vehicle navigation research. Compared to a variety of other sensors, such as LiDAR, ultrasonic and millimeter-wave radar, visual sensors are low-cost and extremely rich in image information [1]. Furthermore, the existing public datasets for lane line detection are mostly visible images. Therefore, there is a wide range of applications for visual navigation in intelligent vehicle navigation. However, vision sensors are susceptible to weather and lighting constraints [2], lane lines are severely obscured by vehicles and pedestrians, as frequently happens in realistic scenarios, and lane lines in the distance are always too blurry to catch the lane line features in deep models.

In response to the above problems, a major research focus has been on designing better network architectures, with little attention paid to data augmentation. Data augmentation techniques are widely used to improve the performance of benchmark datasets, increasing the quantity and diversity of the training dataset. While most deep learning frameworks implement basic image transformations in the pre-processing of data stage, these transformation operations are usually limited to some variants and combinations of flipping, rotating, scaling, and cropping [3]. However, data augmentation methods are not necessarily universal for different tasks and application scenes, and different data augmentation strategies can lead to different performance variations of the model [4–6]. For example, image rotation is an effective data augmentation method on CIFAR-10 but can have a significant negative impact on the MNIST handwritten dataset when the network distinguishes between handwritten digits 6 and 9 [6]. Therefore, there should be suitable data augmentation methods for different task requirements.

The target object of the lane line detection is the lane line with a slender appearance structure. Moreover, the states of the lane lines present uncertainty, such as obscured, worn, and discontinuity when the road changes itself. Most of the current lane line detection models are trained on raw datasets without tailor-made pre-processing of the data. Generally, the raw datasets do not provide enough images with interference conditions. In the face of complex environments, such as vehicles and pedestrians blocking lane lines, water in potholes on the road, trees or buildings shadows on both sides of the road, the accuracy and robustness of the lane line detection model will be negatively affected. Existing data augmentation methods with pervasive applicability basically fail to achieve a positive augmentation effect for special lane lines. Therefore, this paper studies a data augmentation framework for the lane line detection problem for complex road environments. Related works are described in Section 1.1 with lane line detection and data augmentation solutions, while Section 1.2 clarifies the paper's contribution.

1.1. Related Works

For the task of lane line detection, how to effectively solve the problems of discontinuous lane lines, the disappearance of lane lines in the distance and blurry lane lines caused by actual road conditions (strong light, shadows, occlusions, broken road surfaces, etc.) is very important. At present, the main methods of solving environmental interference are improving the detection algorithm and enhancing the lane line images in the datasets.

1.1.1. Lane Line Detection

Currently, many scholars are dedicated to improving lane line detection algorithms. Previous methods are based on hand-crafted low-level features [7]. However, to cope with complex road scenes, designing network models based on deep learning is the current mainstream technical solution [8]. Deep learning-based lane line detection can be summarized into three categories, depending on the representation of the lane lines.

Segmentation-based lane line detection. Pixel-based prediction method classifies each pixel as a lane line or background. Spatial CNN (SCNN) transforms the traditional deep layer-by-layer convolutions into slice-by-slice convolutions in the feature map [9],

which is suitable for detecting long-distance continuous shape targets with extremely strong spatial relationships. Hou et al., proposed self-attentive distillation (SAD) [10], which allows the model to learn from itself, guaranteeing real-time efficiency while achieving high performance without any additional supervision or labeling. Furthermore, a network of teachers and students was used to convey structural relationships between lane lines by constructing inter-regional affinity distillation maps [11]. Zheng et al., continued the idea of SCNN and went on to use a novel module named Recurrent feature-shift aggregator to convey information more efficiently [12]. However, algorithms based on the idea of semantic segmentation use only local features and may miss parts of the lane lines, so these algorithms may not be able to detect inconspicuous lane lines in complex scenes.

Anchor-based lane line detection. Line-CNN, proposed by Li et al., is a pioneering work on the use of line proposal units in lane line detection [13]. Attention-guided lane detection (LaneATT) was proposed with a new anchor-based attention mechanism for lane line detection [14]. Qin et al., first proposed a row-anchor-based lane line detection method that converts lane line detection into a row classification task [15], achieving the fastest lane line detection speed available. CondLaneNet introduces a conditional lane line detection strategy based on conditional convolution and a row-anchor-based representation [16]. However, in some complex scenes, the starting point of lane lines is often difficult to identify, which leads to relatively poor detection performance of the model.

Parametric regression-based lane line detection. Wang et al., proposed a three-branch network to regress the polynomial coefficients on each lane line and estimate its starting and ending points [17]. Tabelini et al., output polynomials to represent each lane marker in the image by polynomial regression [18]. Long short-term transformer (LSTR) considers the road structure together with the camera pose to model the shape of the lane lines [19]. Feng et al., proposed a lane line detection model based on 3rd-order Bessel curves to improve the curve fitting ability [20]. Jin et al., introduced the concept of feature lanes to determine the optimal set of lanes by clustering the training lanes into the feature lane space and generating a set of candidate lanes [21]. Current algorithms for lane line detection do not propose tailor-made data pre-processing methods for interference problems such as shadows, occlusions, and glare and do not propose substantial solutions to environmental interference problems.

1.1.2. Data Augmentation

Data augmentation is an effective method to solve the problem of limited training data for deep learning models. Most data augmentation methods have been proposed to augment the training dataset and improve the generalization ability of neural networks.

Single-sample image space data augmentation. Common image augmentation methods are based on image transformations such as flipping, rotating, cropping, dithering, and blurring [22–26]. Zhong et al., proposed random erasing to achieve augmentation of the training image set for deep learning [27]. The main implementation of this method is to perform a rectangular crop of the image and then replace the pixels within the rectangular area with random values. Unlike random erasing, cutout considers the area of the erased region to be more important than the shape [28], and the erased region is not required to be a rectangle or other regularized shape. Also, for filling the erased region, cutout advocates using a 0 mask for filling rather than using random noise. Another similar research idea is the DropRegion data augmentation applied to Chinese character recognition [29].

Synthetic-sample image space data augmentation. As the number of layers of deep neural networks continues to expand and the expressive power of models continues to increase, synthetic sample data augmentation methods [30–33], represented by mixup [34], have emerged to better prevent model overfitting. Guo et al., argued that the effectiveness of pixel-blending image augmentation, such as mixup, has not been fully demonstrated [35]. To this end, Guo et al., proposed adversarial domain adaptation with domain mixup (AdaMixup) as an adaptive version and considered the method as a hybrid strategy for adaptively learning mixup with regularization techniques outside of neural networks [35].

To further improve the performance of blended sample image augmentation, Summers et al., proposed a multiple non-linear blending method [36]. Other derivative studies about mixup include feature layer mixing [37], manifold mixup [38], etc. Random image cropping and patching (RICAP) is a novel image augmentation method proposed by Takahashi et al. [39]. The idea of RICAP is to randomly select four samples of images, crop a portion of each and put them together to form a new sample. Inspired by the ideas of cutout image masking and mixup image blending, Yun et al., proposed a cut-and-mix image augmentation method [40]. Cut, paste and learn is an effective image augmentation method for target detection, as proposed by Dwibedi et al. [41]. The idea of this method is to randomly select regions of example pixels and superimpose them randomly onto the background images.

Feature space data augmentation. Devries and Taylor proposed a scheme for data augmentation in the dataset feature space [42]. Liu et al., argued that data augmentation methods in image space produced very limited reasonable data and proposed an adversarial autoencoder image augmentation method with linear interpolation [43]. Research has shown that generative adversarial networks (GANs) are an effective unsupervised method for image data augmentation [44–46]. To be able to provide more auxiliary information for semi-supervised training, Odena et al., proposed an auxiliary classifier GAN (ACGAN) [47], which adds an additional classification task to the discriminator of conditional generative adversarial networks. Huang et al., proposed actor-critic GAN based on the ACGAN to address the problem of intra-class data imbalance in image classification applications [48]. Singh et al., proposed malware image synthesis using GANs based on the ACGAN to solve the problem of lack of image data with labels [49]. Tran et al., proposed a bayesian data augmentation approach based on GANs [50], for which the sample generation module can be flexibly replaced with a different generative network model. In summary, data augmentation in feature space is as effective as the image space augmentation methods described above.

1.1.3. Data Augmentation for the Lane Line Detection

Strategies and approaches to data augmentation vary across application datasets and scenes. Therefore, the best data augmentation method needs to be found for a specific image dataset and application scene. However, these data augmentation methods often fail to achieve good augmentation results when applied to lane line detection. For lane line detection, Gu et al., proposed an improved color balance data augmentation algorithm based on the adaptive local tone mapping (ALTM) algorithm [51], which facilitates the extraction of lane line features. Zheng et al., used a generative adversarial network-based data augmentation method to convert the daytime environment into a nighttime environment in the same scene to improve the accuracy of lane line detection in low-light environments [52]. Liu et al., improved the RandAugment data augmentation algorithm and then applied it to lane line detection and obtained an enhancement effect [53]. There are very few data augmentation methods for lane line detection in existing research, especially for special occlusion scenes. Therefore, it makes sense to design a realistic scene-oriented data augmentation framework for lane line detection.

1.2. Paper Contribution

We propose a dynamic data augmentation framework to alleviate the problems faced in open-world lane line detection. By dynamically simulating the occlusion and low visibility existing in the real road environment, the road conditions in the open datasets are made closer to the real environment. Furthermore, the data-enhanced images are dynamically trained in each batch to improve detection accuracy. This paper provides the following contributions:

- A dynamic data augmentation framework based on imitating real scene is proposed. The framework can be integrated with a variety of training-based models without changing the learning strategy, any additional parameter learning or memory con-

sumption, and is a lightweight and plug-and-play framework. It complements existing data augmentation methods for lane line detection.

- Three dynamic data augmentation strategies that simulate different realistic scenes are contained in the framework. Different simulation styles are added to the dynamically selected training dataset in different ways to simulate the three scenes of crowded, shadow and dazzle. Experiment results show that our strategies can improve the robustness of the lane line detection model to detect partially obscured samples. For example, the lane lines in the long distance are effectively extended in the test results.

2. Methods

The proposed framework contains three dynamic data augmentation strategies for simulating different scenes: dynamic simulate shadows (DSS), dynamic simulate highlight (DSH), and dynamic simulate occlusion (DSO). Through the above strategies, different effects are added to the training images to solve the problems of discontinuous, inaccurate and blurry detection results. Figure 1 shows an overview of the proposed framework. First, some images are selected in the training dataset with a random probability. The DSS adds shadows of different sizes and shapes dynamically to both sides of the selected images to simulate a situation where the lane lines are obscured by the shadows of trees or buildings on the sides of the road. Second, a different number of training images are selected in a similar way. The DSH adds several elliptical spots of different sizes with a dazzling effect dynamically to specific areas of the selected images to simulate the reflected brightness of potholes or strong light at the far end of the road. Third, a portion of the training images is selected in a similar way for the next image processing operation. The DSO adds square shapes of different scales and areas dynamically at reasonable locations in the selected images, which are filled with adaptive pixel values, to simulate the situation where the lane lines are blocked by vehicles and pedestrians. Our proposed framework dynamically uses three data augmentation strategies in each epoch of the training process. The probability of randomly selecting images in each epoch varies among strategies.

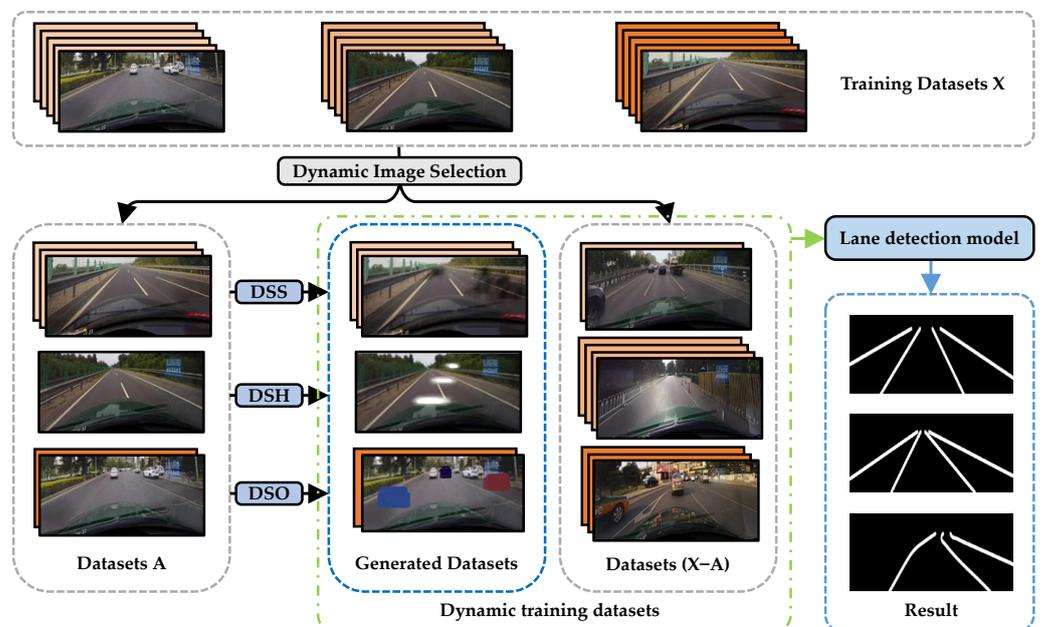


Figure 1. Overview of the proposed framework.

2.1. DDA-IRS-DSS

In the training images of the existing dataset, most of the images are captured on wide roads without trees or buildings on both sides. However, in real complex road environments, there are shadows on the road surface caused by cars, trees and buildings.

To improve the effect of shadows on lane line features in such scenes, we propose the DSS dynamically simulates the shadows on the sides of the road. First, some of the training images are selected adaptively during the training process. Then, random shadows are added to the selected images to extend the original dataset. In this way, we obtain training images adapted to the shaded scenes. Algorithm 1 shows the dynamic simulation of road shadows procedure.

In training, the shadow content in the image is analyzed by histogram. Random shadows are added with probability p_s at reasonable locations in the images where the shadow content is below a threshold. The shadows are superimposed on the original image as a translucent black mask. That is, the probability that the image will remain unchanged is $1 - p_s$. Specifically, we suppose the size of the training image is $W \times H$, and the area of the image is $S = W \times H$. We initialize the area of the random shadow mask to S_s , where S_s/S is within the range specified by the minimum S_l and the maximum S_h . A random point is selected in the edge area on both sides of the road of the image, and its coordinates are set to (x_s, y_s) , which satisfies $(0 \leq x_s \leq W/3$ and $x_s \leq y_s \leq 3 \times H/W \times x_s)$ or $(W \times 2/3 \leq x_s \leq W$ and $W - x_s \leq y_s \leq 3 \times H/W \times (W - x_s))$. We adaptively resize the range of S_l and S_h by the value of the vertical coordinate y_s to ensure that the further away the shadow mask is, the smaller its size will be. A randomly shaped shadow of the area S_s containing n vertices is made clockwise with (x_s, y_s) as the starting point. We set this shadow area as the mask area and use adaptive semi-transparent black as the mask color.

Algorithm 1: Dynamic Simulation of Road Shadows Procedure

Input: Input image I ;
Image size W and H ;
Area of image S ;
Simulating road shadows probability p_s ;
Simulating road shadows area ratio range S_l and S_h ;
Number of vertices n .

Output: Simulated image I^* .

Initialization: $p_1 \leftarrow \text{Rand}(0, 1)$.

- 1 **if** $p_1 \geq p_s$ **then**
- 2 $I^* \leftarrow I$;
- 3 **return** I^* .
- 4 **else**
- 5 **while** *True* **do**
- 6 $S_s \leftarrow \text{Rand}(S_l, S_h) \times S$;
- 7 $x_s \leftarrow \text{Rand}(0, W/3)$ and $y_s \leftarrow \text{Rand}(x_s, 3 \times H/W \times x_s)$ or
 $x_s \leftarrow \text{Rand}(W \times 2/3, W)$ and $y_s \leftarrow \text{Rand}(W - x_s, 3 \times H/W \times (W - x_s))$;
- 8 $I_s \leftarrow (x_s, y_s, n, S_s)$;
- 9 $I_s \leftarrow (0, 0, 0)$;
- 10 $I^* \leftarrow \text{pro}_1 \times I_s + \text{pro}_2 \times I$, where $\text{pro}_1 + \text{pro}_2 = 1$;
- 11 **return** I^* .
- 12 **end**
- 13 **end**

The degree of translucency of the black shadows varies from image to image. Dynamic data augmentation is achieved by adding shadows of different shapes, areas and transparency in different images. Overlaying the mask on the original image preserves the features of this area of the original image to achieve a realistic shadow effect.

2.2. DDA-IRS-DSH

For the real scene of lane line detection, we propose DSH to simulate the light spots caused by sunlight reflections from the road puddles or the shooting camera, too much light at the end of the road, etc. We dynamically acquire a part of the training images in each epoch in a random way and dynamically add oval light spots with dazzling effects

at reasonable locations on the image. Algorithm 2 shows the dynamic simulation of highlights procedure.

Algorithm 2: Dynamic Simulation of Highlights Procedure

Input: Input image I ;
Image size W and H ;
Simulating highlights probability p_h ;

Output: Simulated image I^* .

Initialization: $p_2 \leftarrow \text{Rand}(0, 1)$.

```

1  if  $p_2 \geq p_h$  then
2     $I^* \leftarrow I$ ;
3  return  $I^*$ .
4  else
5    while True do
6       $x_h \leftarrow \text{Rand}(W/3, W \times 2/3)$ ,  $y_h \leftarrow \text{Rand}(H/4, H/2)$ ;
7       $axesl \leftarrow \text{Rand}(a, b)$ ,  $axess \leftarrow \text{Rand}(c, d)$ , where  $a, b, c, d$  depend on  $y_h$ ;
8      if  $x_h + axesl/2 \leq W$  &  $x_h - axesl/2 \geq 0$  &  $y_h + axess/2 \leq H$  &  $y_h - axess/2 \geq 0$  then
9         $I_h \leftarrow (x_h, y_h, axesl, axess)$ ;
10        $strength \leftarrow \text{Rand}(250, 350)$ ;
11        $radius \leftarrow \max(axesl, axess)$ ;
12        $d \leftarrow \sqrt{(x - x_h)^2 + (y - y_h)^2}$ , where  $(x, y)$  is in the  $I_h$ ;
13        $V \leftarrow strength \times (1 - \sqrt{d/radius})$ ;
14        $R(I_h) \leftarrow \min(255, \max((R(I) + V), 0))$ ;
15        $G(I_h) \leftarrow \min(255, \max((G(I) + V), 0))$ ;
16        $B(I_h) \leftarrow \min(255, \max((B(I) + V), 0))$ ;
17        $I^* \leftarrow pro \times I_h + I$ , where  $pro \leftarrow \text{Rand}(0.3, 0.7)$ ;
18     return  $I^*$ .
19   end
20 end
21 end

```

In order to adjust the dynamic effect of the data augmentation, the operation of adding an elliptical spot with a dazzling effect is performed with a certain probability in training. For n training images in an epoch, $n \times p_h$ training images are selected in a random way with probability p_h . The operation of adding an elliptical spot with a dazzling effect is done in the selected training images. That is, the probability that the image is not modified in any way is $1 - p$. Assuming that the size of the training image is $W \times H$, the DSH selects a region $H/4 \leq h \leq H/2$ and $W/3 \leq w \leq W \times 2/3$, chooses a random coordinate (x_h, y_h) within this region, makes an ellipse region I_h with this coordinate as the central point, and makes a flare effect within this ellipse. The long axis of the ellipse is initialized as $axesl$ and the short axis is initialized as $axess$. The values of both the long and short axes are taken as random numbers within a certain range, where the upper and lower values of this range are related to the coordinates (x_h, y_h) . For a more realistic dazzle of light, then the different positions correspond to different sizes of the ellipse. We adaptively adjust the range of values for the long and short axes in terms of the size of the vertical coordinate y_h , ensuring that the more distant the ellipsoidal spot is, the smaller the size. If $x_h + axesl/2 \leq W$, $y_h + axess/2 \leq H$, $x_h - axesl/2 \geq 0$ and $y_h - axess/2 \geq 0$, we set the region, $I_h = (x_h, y_h, axesl, axess)$, as the selected ellipse region. In addition, the angle of the ellipse is also randomized, with most images guaranteed to have an ellipse angle of 90 degrees. For the fill values of the pixels within the ellipse, we first use a semi-transparent white effect for the pixel fill. In this way, a certain degree of original image features can be

preserved. Next, we adjust the level of glare according to the distance of the pixel point within the ellipse from the central point of the ellipse:

$$V = strength \times \left(1 - \sqrt{d/radius}\right) \quad (1)$$

where V denotes the value of pixels to be added for different pixel points, $strength$ denotes the intensity value of the dazzle presented, d denotes the distance of the pixel point from the central point of the ellipse and $radius$ is the larger between $axes_l$ and $axes_s$.

Once the value V has been obtained, the value V is added to three channels of each pixel in the ellipse, which gives this pixel a translucent effect. Each channel compares the summed value with 255, the smaller number being retained in this channel for that pixel, in this way preventing the pixel value from crossing the boundary.

2.3. DDA-IRS-DSO

Random erasing [27] and cutout [28] are used in image processing for data augmentation in various scenes. Based on the idea of such data augmentation methods, we propose a strategy to simulate the congestion of vehicles, pedestrians, buildings, etc., in real road scenes, by dynamically acquiring part of training images in a random way and dynamically adding adaptive boxes to the acquired images. The distribution of vehicles on the image is used to determine the size of the added boxes. The colors and shapes of the interior of the added boxes depend on the colors and outlines of the real vehicles. Algorithm 3 shows the dynamic simulation of road vehicles procedure.

Algorithm 3: Dynamic Simulation of Road Vehicles Procedure

Input: Input image I ;
Image size W and H ;
Area of image S ;
Simulating road vehicles probability p_o ;
Simulating road vehicles area ratio range S_l and S_h ;
Simulating road vehicles aspect ratio range r_1 and r_2 .

Output: Simulated image I^* .

Initialization: $p_3 \leftarrow \text{Rand}(0, 1)$.

- 1 **if** $p_3 \geq p_o$ **then**
- 2 $I^* \leftarrow I$;
- 3 **return** I^* .
- 4 **else**
- 5 **while** *True* **do**
- 6 $x_o \leftarrow \text{Rand}(0, W)$, $y_o \leftarrow \text{Rand}(0, H)$;
- 7 $S_o \leftarrow \text{Rand}(S_l, S_h) \times S$, where S_l and S_h depend on y_o ;
- 8 $r_o \leftarrow \text{Rand}(r_1, r_2)$;
- 9 $H_o = \sqrt{S_o \times r_o}$, $W_o = \sqrt{S_o / r_o}$;
- 10 **if** $x_o + W_o \leq W$ and $y_o + H_o \leq H$ **then**
- 11 $I_o = (x_o, y_o, x_o + W_o, y_o + H_o)$, $I(I_o) \leftarrow \text{Random}$;
- 12 $I^* \leftarrow I$;
- 13 **return** I^* .
- 14 **end**
- 15 **end**
- 16 **end**

A count of all types of vehicles on the road from the driver's point of view, counting their shape, color and characteristics in order to simulate a more realistic scene of vehicle congestion on the road. In training, simulated vehicles are added with a certain probability in order to achieve a dynamic addition effect. For an image I in a batch, the probability that it will do the simulation of adding a vehicle is p_o , and the probability that it will be held constant is $1 - p_o$. The DSO selects a random coordinate (x_o, y_o) in the image, makes

a rectangular region I_o of area initialized by S_o with this coordinate as the top left point, and overlays the original rectangular region with a set value. The coordinates (x_o, y_o) are set in the lower middle region of the image to avoid the simulated vehicles appearing in unreasonable locations, such as the sky or the sides of the road. The size of the training image is assumed to be $W \times H$, and the area of the image is $S = W \times H$. S_o/S is in the range between the minimum S_l and the maximum S_h , where the values of S_l and S_h are related to the coordinates (x_o, y_o) . In order to simulate road vehicles more realistically, then the location varies corresponding to the size of the vehicle. We adaptively adjust the size of the rectangular area S_o by the size of the vertical coordinate y_o to ensure that the size of the rectangle is smaller the more distant it is while satisfying the various types of vehicles at the time of the count. The aspect ratio of the rectangular area is initialized randomly between r_1 and r_2 , and we set the initialized aspect ratio to r_o . The length of the rectangular area is $H_o = \sqrt{S_o \times r_o}$ and the width is $W_o = \sqrt{S_o/r_o}$. If $x_o + W_o \leq W$ and $y_o + H_o \leq H$, we set the region, $I_o = (x_o, y_o, x_o + W_o, y_o + H_o)$, as the selected rectangular region. Each pixel value in I_o is adaptively aligned with the statistically derived pixel value of the simulated real vehicle.

3. Experimental Results

3.1. Datasets

CULane [9]: This is a generic dataset for lane line detection and contains approximately 130 k images. The dataset is divided into 88,880 training, 9675 validation, and 34,680 testing images. Of these, the testing images are divided into a “Normal” category and eight challenging categories, including “Crowded”, “Night”, “No line”, “Shadow”, “Arrow”, “Dazzle”, “Curve”, and “Crossroad”. For some categories, the lane lines are heavily obscured and even invisible. For each image, there are up to four-lane lines labeled. Most of the dataset is straight lanes, with few curved roads.

3.2. Evaluation Metrics

For the CULane dataset, each lane line is treated as a thin line of equal width of 30 pixels. Intersection-over-union (IoU) between the predicted result and the ground truth is calculated. If the value of IoU is greater than a threshold of 0.5, the predicted lane lines are considered true positives (TP). The precision and the recall are computed by

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2)$$

where TP is the number of correctly detected lanes, FP is the number of false positives, and FN is that of false negatives. Then, the F-measure is computed by

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

3.3. Experimental Settings

To evaluate the effectiveness of the proposed DDA-IRS, we compare the detection results of DDA-IRS with several representative methods. All the networks in this article are implemented on the Pytorch platform. The equipment used in the following experiments consists of an Intel (R) Core (TM) CPU i5-9400F @ 2.90 GHz and one NVIDIA GeForce RTX 2060 GPU with 6 GB of memory. We choose ResNet-18 as the backbone for experiments. We train 300 epochs for CULane with a batchsize of 2 per GPU. The proposed DDA-IRS is applied to the training phase, including DSS with a probability of 0.4, DSH with a probability of 0.3, and DSO with a probability of 0.2.

3.4. Comparative Assessment

Table 1 compares the F-measure performances on the CULane dataset and shows that the model with data augmentation by DDA-IRS outperforms the baseline model and all

traditional algorithms. Our DDA-IRS achieves the state-of-the-art result on the CULane dataset with 76.5% F-measure. The data-enhanced model outperforms the baseline model by 0.5% in terms of F-measure. And the F-measure under the “Normal”, “Crowded”, “Shadow”, “Arrow”, and “Curve” categories achieves a 0.4%, 0.1%, 1.6%, 0.4% and 1.4% improvement, respectively. Besides, the “Crossroad” category in the evaluation metrics of the CULane dataset does not calculate the values of TP, Precision and Recall but only evaluates the performance of this category by the value of FP. The smaller the value of FP, the better the performance in this category. Compared with the base model, although our framework performs better, only 0.5% of F1-measure since CULane is a simpler dataset with many straight lines, it has considerable improvements in “Shadow” and “Curve” scenes, which demonstrates that imitating real scenes can strengthen local location connectivity. This shows that DDA-IRS purposefully improves the detection performance under occlusion in realistic scenes. For the challenging categories of “Night”, “No line”, and “Dazzle”, the F-measure drops slightly or stays the same. However, it is not that our method is not applicable to these scenes, but rather that the testing images for these scenes have a partial labeling bias as well as contain a large number of non-labeled images. The correct detection results for such test images cannot be classified as TP, leading to a decrease in values when calculating the F-measure.

To analyze this performance degradation, we visualized the recognition results for these three categories, as shown in Figure 2. The visualization results show that DDA-IRS identifies the lane lines more completely compared to the baseline in all three categories, i.e., “Night”, “No line”, and “Dazzle”. Meanwhile, we can see that the labels are missing or incomplete, while the data-enhanced model can detect the unlabeled lane lines and effectively extend lane lines that are incompletely labeled. Therefore, although the detection results obtained are correct, the calculation of the IoU will result in a calculated result that is less than the threshold value. Then, such a test result cannot be classified as TP, and the value of the F-measure obtained in the evaluation metric is slightly reduced. This does not mean that DDA-IRS is ineffective in these scenes. On the contrary, DDA-IRS is equally effective in the “Night”, “No line”, and “Dazzle” categories. By DDA-IRS, clearer and more complete results are still obtained under these categories, and there are fewer missed detections.

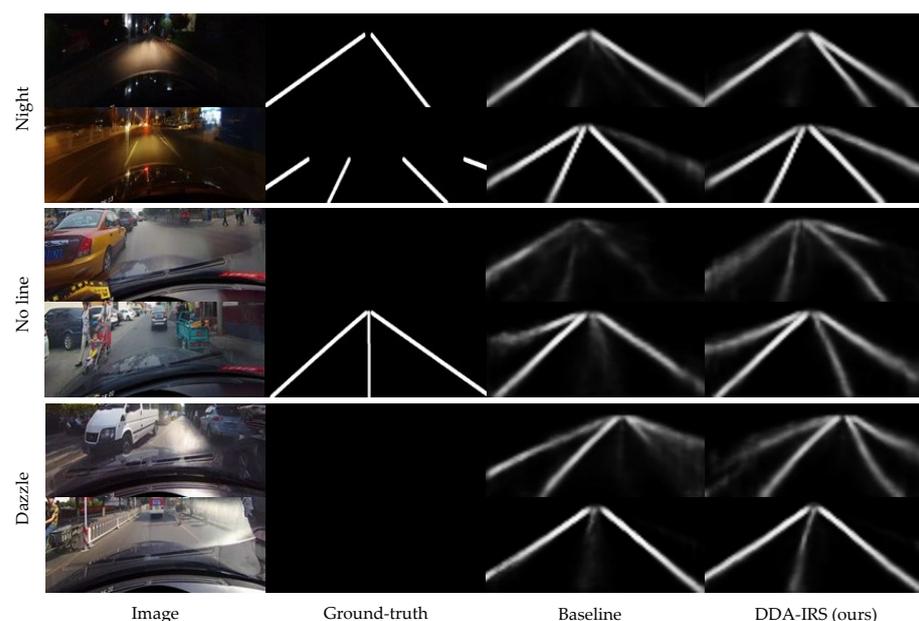


Figure 2. Comparison of lane detection results for inaccurately labeled or unlabeled lane lines on three challenging categories in the CULane dataset.

Table 1. Comparison of the F-measure performances (%) on the CULane dataset with other detection models. $\hat{\cdot}$ means that the encoder backbone is ResNet18.

Category	Normal	Crowded	Night	No Line	Shadow	Arrow	Dazzle	Curve	Crossroad	Total
SCNN [9]	90.6	69.7	66.1	43.4	66.9	84.1	58.5	65.7	1990	71.6
SAD [10]	90.7	70.0	66.3	43.5	67.0	84.4	59.9	65.7	2052	71.8
Curve-Nas [54]	90.7	72.3	68.9	49.4	70.1	85.8	67.7	68.4	1746	74.8
LaneATT $\hat{}$ [14]	91.1	73.0	69.0	48.4	70.9	85.5	65.7	63.4	1170	75.1
UFast $\hat{}$ [15]	87.7	66.0	62.1	40.2	62.8	81.0	58.4	57.9	1743	68.4
Baseline $\hat{}$ [21]	91.1	74.7	69.5	50.9	71.8	87.3	69.8	60.8	1568	76.0
DDA-IRS (ours) $\hat{}$	91.5	74.8	69.4	49.5	73.4	87.7	69.8	62.2	1350	76.5

For the “Crossroad” category, only FP is reported.

In addition, two data augmentation methods for lane line detection are replicated on the baseline model [52,54]. Table 2 compares the F-measure performance of the three models on the CULane dataset. The results show that DDA-IRS outperforms the other two data augmentation methods, particularly achieving better detection results on the challenging categories of “Crowded”, “Shadow”, “Dazzle”, “Curve”, and “Crossroad”. Compared to the better F-measure of the other two data augmentation methods under each category, the proposed DDA-IRS is still 0.2%, 0.8%, 1.9%, 1.1%, and 0.8% higher under “Normal”, “Crowded”, “Shadow”, “Dazzle”, and “Curve” respectively. Of these, “Crowded”, “Shadow”, and “Dazzle” are the main scenes simulated by DDA-IRS, and the lane lines in these scenes are highly implicit or even invisible, demonstrating the effectiveness of DDA-IRS.

Figure 3 compares the visual detection results in several scenes. In the “Crowded”, “Shadow”, and “Dazzle” scenes, the data-enhanced models by other data augmentation methods still fail to detect the highly occluded lane lines. At the same time, DDA-IRS can detect these lane lines clearly and completely. Moreover, in the “Crowded” scene, the data-enhanced models using other data augmentation methods produce misdetection, which incorrectly detects the line of sunlight on the ground as a lane line, while the data-enhanced model using the proposed DDA-IRS does not produce misdetection. This shows that DDA-IRS provides a significant improvement in lane line detection, giving clearer and more complete detection results in most scenes.

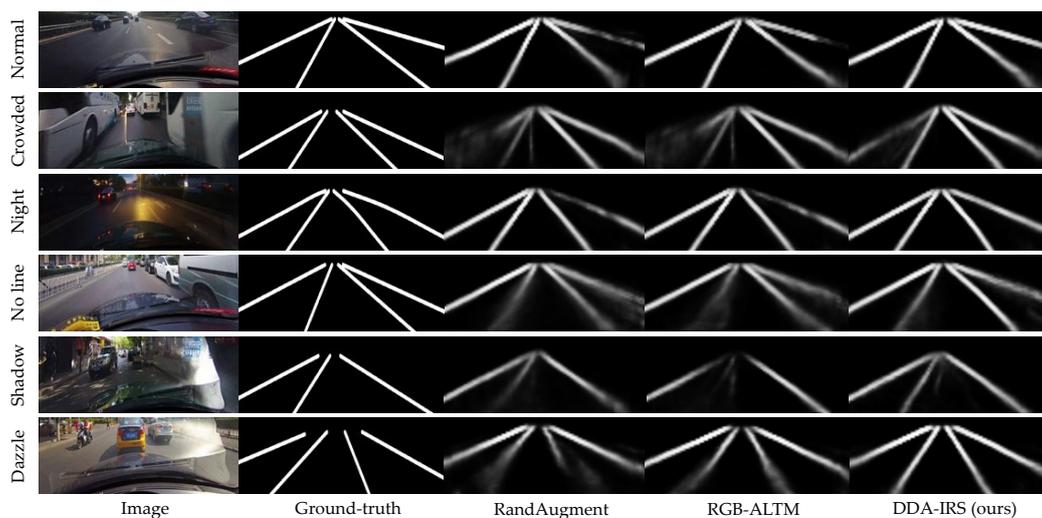
**Figure 3.** Comparison of lane detection results on six categories in the CULane dataset.

Table 2. Comparison of the F-measure performances (%) on the CULane dataset with other data augmentation methods.

Category	Normal	Crowded	Night	No Line	Shadow	Arrow	Dazzle	Curve	Crossroad	Total
Baseline [21]	91.1	74.7	69.5	50.9	71.8	87.3	69.8	60.8	1568	76.0
RandAugment [53]	91.1	74.0	69.4	50.5	69.5	87.3	68.7	61.4	1404	75.8
RGB-ALTM [51]	91.3	74.0	70.1	50.9	71.5	87.7	68.6	59.7	1468	75.9
DDA-IRS (ours)	91.5	74.8	69.4	49.5	73.4	87.7	69.8	62.2	1350	76.5

For the “Crossroad” category, only FP is reported.

3.5. Ablation Studies

We conduct ablation studies to analyze the efficacy of the three strategies of DDA-IRS on the CULane dataset. Table 3 compares the F-measure performances between the baseline model and the data-enhanced model with each of the proposed strategies. It can be seen that using each of the strategies individually improves F-measure performance in general.

Table 3. Comparison of the F-measure performances (%) on the CULane dataset of each proposed strategy.

Category	Normal	Crowded	Night	No Line	Shadow	Arrow	Dazzle	Curve	Crossroad	Total
Baseline [21]	91.1	74.7	69.5	50.9	71.8	87.3	69.8	60.8	1568	76.0
Proposed-DSS	91.6	74.3	70.5	50.3	71.9	87.3	69.4	62.1	1291	76.3
Proposed-DSH	91.3	74.4	71.0	50.4	70.1	87.3	70.0	63.7	1526	76.2
Proposed-DSO	91.3	73.8	70.0	50.6	72.4	87.0	69.6	60.2	1263	76.1

For the “Crossroad” category, only FP is reported.

Efficacy of DSS: Integrity and correctness are very important for lane line detection. “Proposed-DSS” in Table 3 indicates the test results of the data-enhanced model using DSS alone. In the experiment, the simulated shadows were added to the training images individually with a probability of 0.4. Table 3 compares the performance of the data-enhanced model using DSS alone with the original baseline model. The results show that the data-enhanced model using DSS has different degrees of improvement for the five categories of “Normal”, “Night”, “Shadow”, “Curve”, and “Crossroad”, with the F-measure improving by 0.5%, 1.0%, 0.1%, and 1.3% for the “Normal”, “Night”, “Shadow” and “Curve”, categories, respectively, while ensuring that the F-measure remains the same or decreases slightly for other categories. Moreover, the value of the integrated F-measure has been improved by 0.3%.

Figure 4 shows the visualization of the test results containing a large number of shadows. In the testing images with a large number of shadows, the lane lines are mostly indistinct or even invisible. The results show that DSS effectively solves the problem of blurry and incomplete lane line detection in shaded scenes. On parts of the lane lines closer to the horizon, the predictions are more accurate, which improves the phenomenon of lane lines in the distance disappearing to a certain extent. Better still, DSS also solves the problem of missed detections that exist in a normal scene to a certain extent. In summary, the DSS strategy is effective in improving lane line detection results, especially for the shaded scene.

Efficacy of DSH: The clarity of the results and the extension of the lane lines in the distance are important for lane line detection. In the experiment, the elliptical patches of light with dazzling effects were added to the training images individually with a probability of 0.3. The results are shown in Table 3 under “Proposed-DSH”. Comparing the proposed DSH and Baseline, the results show that the data-enhanced model using DSH has different degrees of improvement for the five categories of “Normal”, “Night”, “Dazzle”, “Curve”, and “Crossroad”, with the F-measure improving by 0.2%, 1.5%, 0.2% and 2.9%, for the “Normal”, “Night”, “Dazzle”, and “Curve” categories respectively, while ensuring that the

F-measure remains the same or decreases slightly for other categories. Moreover, the value of the F-measure is also improved in general by 0.2%.

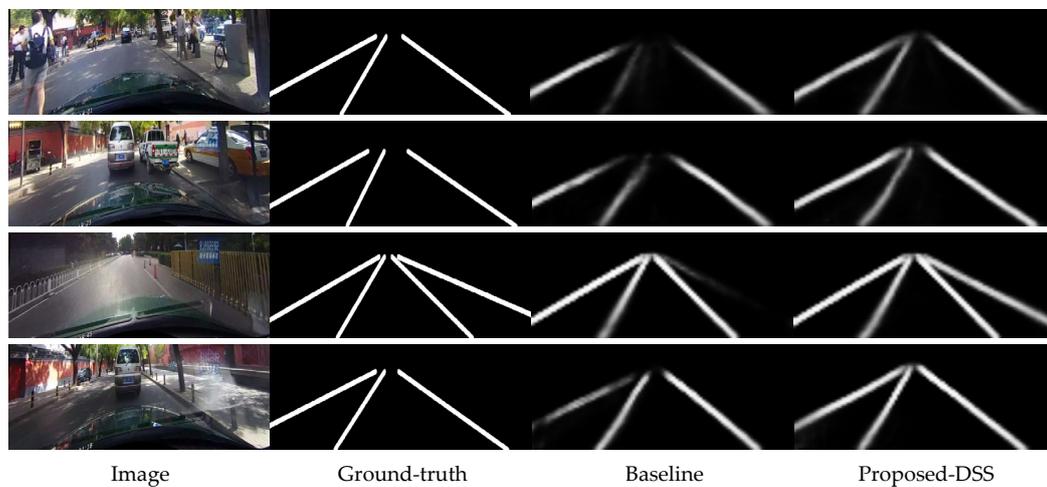


Figure 4. Comparison of lane detection results for the shadow scene on the CULane dataset.

In addition, a visual comparison of the test results for images containing dazzling lights in several different scenes is presented in Figure 5. The lights of vehicles at night and the glare of the sun during the day can cause lane lines that would be clearly visible to become invisible. The results show that DSH effectively solves the problem of blurry and incomplete test results in a dazzling light scene to a certain extent. Moreover, DSH effectively improves the disappearance of lane lines in the distance.

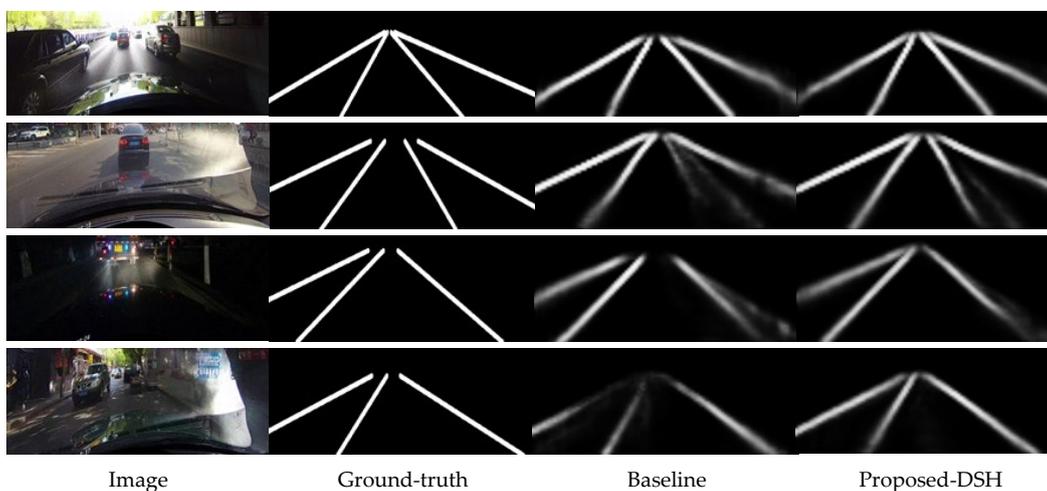


Figure 5. Comparison of lane detection results on the CULane dataset with dazzling light.

Efficacy of DSO: For road environments where lane lines are heavily obscured, such as congestion, it is important to detect lane lines under obscuration. To verify the performance of DSO, we added multi-size boxes to the training images to enhance the data with a probability of 0.2 individually and recorded the experimental results in Table 3 under “Proposed-DSO”. A comparison of the proposed DSO and baseline shows that the lane line labels are usually not accurate enough in the testing images with heavy occlusion. The data-enhanced model with DSH has different degrees of improvement for the four categories of “Normal”, “Night”, “Shadow,” and “Crossroad”, with the F-measure improving by 0.2%, 0.5% and 0.6% for the “Normal”, “Night”, and “Shadow” categories respectively, while ensuring that the F-measure remains the same or decreases slightly for the other

categories. Moreover, the value of the F-measure is improved in general by 0.1%. The drop in F-measure for the “Crowded” category is explained in the visualization results.

Figure 6 shows a visual comparison of the test results for the congested environment. It can be seen that the lane line labels are usually not accurate enough in the testing images with heavy occlusion. Our strategy allows for more effective detection of obscured lane lines in crowded environments and extends detection results of lane lines in the distance. Furthermore, as can be seen in the figure, the data-enhanced model can detect lane lines that are not represented in the ground truth, such that the result can lead to less than a threshold of 0.5 when calculating the IoU between the detection result and the ground-truth, resulting in correctly predicted lane lines not being recognized as TP, making both Precision and Recall smaller, which in turn leads to a decrease in F-measure. However, it can be seen that the detection results for the “Crowded” category with DSO are actually better than the visualization results. In conclusion, the DSO strategy in DDA-IRS can effectively improve lane line detection in occlusion scenes.

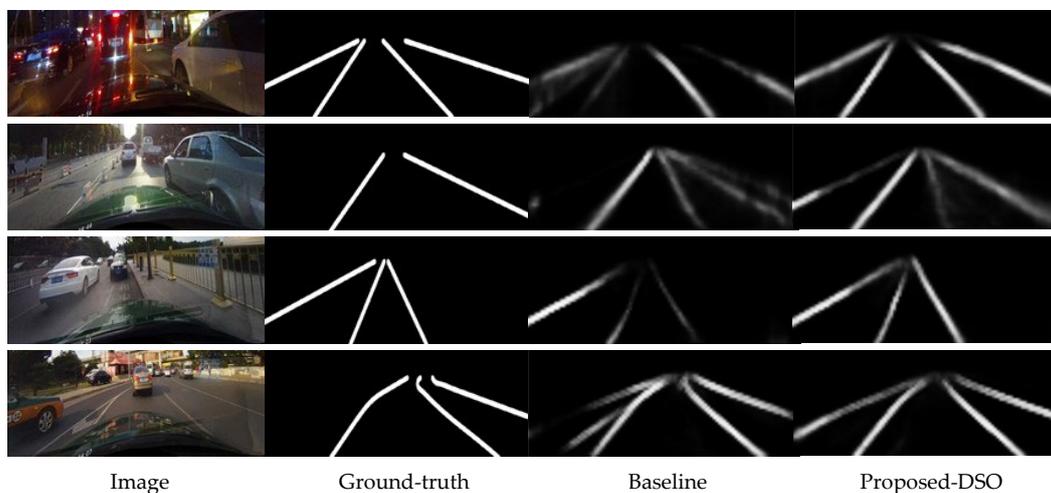


Figure 6. Comparison of lane detection results for the crowded scene on the CULane dataset.

4. Discussion

We propose a data augmentation framework for lane line detection that improves detection when oriented to realistic road scenes (e.g., shadows, reflections, occlusions, etc.). We considered these three most common realistic road scenes that bring about the problem of lane lines becoming invisible. Some other complex scenarios, such as low light and adverse meteorological conditions (e.g., snow or ice on the road, heavy snowfall, heavy fog, etc.), were not fully taken into account for the framework. Such scenes would likewise present highly implicit or even invisible lane lines. This leaves our method that may not have sufficient generalization capabilities. It is also our future research direction to extend the scenes targeted more diverse for lane line detection. In addition, when facing the occlusions scene, we adopt the strategy of adding multiple boxes of different colors and sizes to simulate the occlusion of vehicles or pedestrians on the road. However, there are still gaps between the pixels of the added regions and the real occlusion colors, and the outlines of the regions are stiff. If the added regions can be closer to the real occlusions in terms of color and outline, the method could achieve better detection effects. Moreover, from the experimental results, we find that in addition to the performance improvement in scenes with occlusions, the F-measure value also increases considerably in the “Curves” category. This enhancement is outside what we are targeting in the scenes. Based on the results, further studies can be designed for the curve scene, which will have research significance.

5. Conclusions

In this paper, we propose a simple and effective dynamic data augmentation framework, named DDA-IRS, to alleviate the problems faced by lane line detection in complex scenes. DDA-IRS does not require any additional parameter learning or memory consumption and can be integrated with various training-based models without modifying the original model, making it a plug-and-play module. DDA-IRS focuses on disturbances in complex scenes and contains three different data augmentation strategies for simulating different realistic road scenes (i.e., shadows, reflections, occlusions). Three different data augmentation strategies add different modules to randomly selected images, simulating shadows through semi-transparent masks of different shapes, reflections through elliptical patches of light with a dazzling effect, and occlusions through multi-size boxes with adaptive pixels. In the model training process, the different data augmentation strategies are randomly applied to achieve a more realistic effect. We apply DDA-IRS to a model with ResNet18 as the backbone network and conduct experiments on the CULane dataset. The results show that DDA-IRS improves the robustness of the lane line detection model to partially occluded samples and extends the detection results at the lane lines in the distance. Surprisingly, we find that DDA-IRS has the ability to improve the detection of lane lines on categories such as curves and intersections. In the future, the application of DDA-IRS to lane lines with structural diversity, such as curves, is worthwhile research work.

Author Contributions: Conceptualization, Q.W. and T.S.; methodology, Q.W.; software, L.W.; validation, Y.C., J.S., J.G. and S.S.; formal analysis, Q.W. and L.W.; investigation, Y.C. and J.S.; resources, T.S.; data curation, L.W.; writing—original draft preparation, L.W.; writing—review and editing, Q.W.; visualization, L.W.; supervision, T.S.; project administration, T.S.; funding acquisition, Q.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Youth Project of the National Natural Science Foundation of China under Grant 62201237, in part by the National Natural Science Foundation under Grant 61971208, in part by the Yunnan Fundamental Research Projects under Grant 202101BE070001-008, in part by the International Research Cooperation Seed Fund of Beijing University of Technology under Grant 2021B04, and in part by the Fundamental Research Project of Yunnan Province under Grant 202101AU070050.

Data Availability Statement: The CULane traffic road inspection dataset is obtained from <https://xingangpan.github.io/projects/CULane.html>, accessed on 2 February 2018.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Son, M.; Chang, H.S. Lane line map estimation for visual alignment. In Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR), Utrecht, The Netherlands, 14–18 December 2020; pp. 200–204.
2. Zhou, H.; Sun, M.; Ren, X.; Wang, X. Visible-Thermal image object detection via the combination of illumination conditions and temperature information. *Remote Sens.* **2021**, *13*, 3656. [[CrossRef](#)]
3. Buslaev, A.; Iglovikov, V.I.; Khvedchenya, E.; Parinov, A.; Druzhinin, M.; Kalinin, A.A. Albumentations: Fast and flexible image augmentations. *Information* **2020**, *11*, 125. [[CrossRef](#)]
4. Ratner, A.J.; Ehrenberg, H.R.; Hussain, Z.; Dunnmon, J.; Re, C. Learning to Compose Domain-Specific Transformations for Data Augmentation. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 3236–3246.
5. Cubuk, E.D.; Zoph, B.; Mané, D.; Vasudevan, V.; Le, Q.V. AutoAugment: Learning augmentation strategies from data. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 113–123.
6. Lemley, J.; Bazrafkan, S.; Corcoran, P. Smart augmentation learning an optimal data augmentation strategy. *IEEE Access.* **2017**, *5*, 5858–5869. [[CrossRef](#)]
7. Yu, B.; Jain, A.K. Lane boundary detection using a multiresolution hough transform. In Proceedings of the International Conference on Image Processing (ICIP), Santa Barbara, CA, USA, 26–29 October 1997; pp. 748–751.
8. Sun, T.-Y.; Tsai, S.-J.; Chan, V. HSI color model based lane-marking detection. In Proceedings of the IEEE Intelligent Transportation Systems Conference, Toronto, ON, Canada, 17–20 September 2006; pp. 1168–1172.

9. Pan, X.G.; Shi, J.P.; Luo, P.; Wang, X.G.; Tang, X.O. Spatial as deep: Spatial CNN for traffic scene understanding. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence/30th Innovative Applications of Artificial Intelligence Conference/8th AAAI Symposium on Educational Advances in Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 7276–7283.
10. Hou, Y.; Ma, Z.; Liu, C.; Loy, C.-C. Learning lightweight lane detection CNNs by self attention distillation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1013–1021.
11. Hou, Y.; Ma, Z.; Liu, C.; Hui, T.-W.; Loy, C.C. Inter-region affinity distillation for road marking segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 12486–12495.
12. Zheng, T.; Fang, H.; Zhang, Y.; Tang, W.J.; Yang, Z.; Liu, H.F.; Cai, D. RESA: Recurrent feature-shift aggregator for lane detection. In Proceedings of the 35th AAAI Conference on Artificial Intelligence/33rd Conference on Innovative Applications of Artificial Intelligence/11th Symposium on Educational Advances in Artificial Intelligence, ELECTR Network, Virtual, 2–9 February 2021; pp. 3547–3554.
13. Li, X.; Li, J.; Hu, X.-L.; Yang, J. Line-CNN: End-to-end traffic line detection with line proposal unit. *IEEE Trans. Intell. Transp. Syst.* **2019**, *21*, 248–258. [[CrossRef](#)]
14. Tabelini, L.; Berriel, R.; Paixão, T.M.; Badue, C.; De Souza, A.F.; Oliveira-Santos, T. Keep your eyes on the lane: Real-time attention-guided lane detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 294–302.
15. Qin, Z.-Q.; Wang, H.Y.; Li, X. Ultra fast structure-aware deep lane detection. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 276–291.
16. Liu, L.Z.; Chen, X.H.; Zhu, S.Y.; Tan, P. CondLaneNet: A top-to-down lane detection framework based on conditional convolution. In Proceedings of the 18th IEEE/CVF International Conference on Computer Vision (ICCV), ELECTR Network, Montreal, BC, Canada, 11–17 October 2021; pp. 3753–3762.
17. Wang, B.; Wang, Z.; Zhang, Y. Polynomial regression network for variable-number lane detection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 719–734.
18. Tabelini, L.; Berriel, R.; Paixao, T.-M.; Badue, C.; De Souza, A.-F.; Oliveira-Santos, T. PolyLANE: Lane estimation via deep polynomial regression. In Proceedings of the 25th International Conference on Pattern Recognition (ICPR), ELECTR Network, Milan, Italy, 10–15 January 2021; pp. 6150–6156.
19. Liu, R.J.; Yuan, Z.J.; Liu, T.; Xiong, Z.L. End-to-end lane shape prediction with transformer. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), ELECTR Network, Virtual, 5–9 January 2021; pp. 3693–3701.
20. Feng, Z.Y.; Guo, S.H.; Tan, X.; Xu, K.; Wang, M.; Ma, L.Z. Rethinking efficient lane detection via curve modeling. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17041–17049.
21. Jin, D.; Park, W.; Jeong, S.-G.; Kwon, H.; Kim, C.-S. Eigenlanes: Data-Driven Lane Descriptors for Structurally Diverse Lanes. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 17142–17150.
22. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Martinez-Gonzalez, P.; Garcia-Rodriguez, J. A survey on deep learning techniques for image and video semantic segmentation. *Appl. Soft Comput.* **2018**, *70*, 41–65. [[CrossRef](#)]
23. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [[CrossRef](#)] [[PubMed](#)]
24. Yu, X.R.; Wu, X.M.; Luo, C.B.; Ren, P. Deep learning in remote sensing scene classification: A data augmentation enhanced convolutional neural network framework. *GIScience Remote Sens.* **2017**, *54*, 741–758. [[CrossRef](#)]
25. Parihar, A.S.; Verma, O.P.; Khanna, C. Fuzzy-Contextual Contrast Enhancement. *IEEE Trans. Image Process.* **2017**, *26*, 1810–1819. [[CrossRef](#)] [[PubMed](#)]
26. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 27–30 June 2015; pp. 770–778.
27. Zhong, Z.; Zheng, L.; Kang, G.; Li, S.; Yang, Y. Random erasing data augmentation. In Proceedings of the 34th AAAI Conference on Artificial Intelligence/32nd Innovative Applications of Artificial Intelligence Conference/10th AAAI Symposium on Educational Advances in Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 13001–13008.
28. DeVries, T.; Taylor, G.-W. An empirical evaluation of deep learning on highway driving. *arXiv* **2017**, arXiv:1708.04552.
29. Huang, S.P.; Zhong, Z.Y.; Jin, L.W.; Zhang, S.Y.; Wang, H.B. Dropregion training of inception font network for high-performance Chinese font recognition. *Pattern Recognit.* **2018**, *77*, 395–411. [[CrossRef](#)]
30. Berthelot, D.; Carlini, N.; Goodfellow, I.; Oliver, A.; Papernot, N.; Raffel, C. MixMatch: A holistic approach to semi-supervised learning. In Proceedings of the Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 5050–5060.
31. Wong, S.C.; Gatt, A.; Stamatescu, V.; McDonnell, M.D. Understanding data augmentation for classification: When to warp? In Proceedings of the 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, QLD, Australia, 30 November–2 December 2016; pp. 1–6.
32. Tokozume, Y.; Ushiku, Y.; Harada, T. Between-class learning for image classification. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 5486–5494.

33. Arazo, E.; Ortego, D.; Albert, P.; O'Connor, N.E.; McGuinness, K. Unsupervised label noise modeling and loss correction. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019.
34. Zhang, H.Y.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. mixup: Beyond empirical risk minimization. In Proceedings of the 6th International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018; pp. 1–13.
35. Guo, H.Y.; Mao, Y.Y.; Zhang, R.C. Mixup as locally linear out-of-manifold regularization. In Proceedings of the 33rd AAAI Conference on Artificial Intelligence/31st Innovative Applications of Artificial Intelligence Conference/9th AAAI Symposium on Educational Advances in Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 3714–3722.
36. Summers, C.; Dinneen, M.J. Improved mixed-example data augmentation. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1262–1270.
37. Oki, H.; Kurita, T. Mixup of feature maps in a hidden layer for training of convolutional neural network. In Proceedings of the 25th International Conference on Neural Information Processing (ICONIP), Siem Reap, Cambodia, 13–16 December 2018; pp. 635–644.
38. Verma, V.; Lamb, A.; Beckham, C.; Najafi, A.; Mitliagkas, I.; Lopez-Paz, D.; Bengio, Y. Manifold mixup: Better representations by interpolating hidden states. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019.
39. Takahashi, R.; Matsubara, T.; Uehara, K. Data augmentation using random image cropping and patching for deep CNNs. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 2917–2931. [[CrossRef](#)]
40. Yun, S.; Han, D.; Joon Oh, S.; Chun, S.; Choe, J.; Yoo, Y. CutMix: Regularization strategy to train strong classifiers with localizable features. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6022–6031.
41. Dwibedi, D.; Misra, I.; Hebert, M. Cut, Paste and Learn: Surprisingly easy synthesis for instance detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1301–1310.
42. Devries, T.; Taylor, G.W. Dataset augmentation in feature space. In Proceedings of the 5th International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
43. Liu, X.F.; Zou, Y.; Kong, L.S.; Diao, Z.H.; Yan, J.L.; Wang, J.; Li, S.T.; Jia, P.; You, J.N. Data augmentation via latent space interpolation for image classification. In Proceedings of the 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 728–733.
44. Han, D.; Liu, Q.; Fan, W. A new image classification method using CNN transfer learning and web data augmentation. *Expert Syst. Appl.* **2018**, *95*, 43–56. [[CrossRef](#)]
45. Liang, X.D.; Hu, Z.T.; Zhang, H.; Gan, C.; Xing, E.P. Recurrent topic-transition GAN for visual paragraph generation. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 3382–3391.
46. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2672–2680.
47. Odena, A.; Olah, C.; Shlens, J. Conditional image synthesis with auxiliary classifier GANs. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 2642–2651.
48. Huang, L.; Lin, K.C.-J.; Tseng, Y.C. Resolving Intra-Class Imbalance for GAN-Based Image Augmentation. In Proceedings of the 2019 IEEE International Conference on Multimedia and Expo (ICME), Shanghai, China, 8–12 July 2019; pp. 970–975.
49. Singh, A.; Dutta, D.; Saha, A. MIGAN: Malware image synthesis using GANs. In Proceedings of the 33rd AAAI Conference on Artificial Intelligence/31st Innovative Applications of Artificial Intelligence Conference/9th AAAI Symposium on Educational Advances in Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 10033–10034.
50. Tran, T.; Pham, T.; Carneiro, G.; Palmer, L.; Reid, I. A Bayesian data augmentation approach for learning deep models. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 2797–2806.
51. Gu, D.-Y.; Wang, N.; Li, W.-C.; Chen, L. Method of lane line detection in low illumination environment based on model fusion. *J. Northeast. Univ. Nat. Sci.* **2021**, *42*, 305–309.
52. Zheng, H.; Cheng, S.; Wang, W.; Zhang, M. A lane line detection method based on residual structure. *J. Zhejiang Univ. Technol.* **2022**, *50*, 365–371.
53. Liu, Q.; Wu, C.; Qiao, D. Application of data augmentation in lane line detection. In Proceedings of the 2020 5th International Conference on Computing, Communication and Security (ICCCS), Patna, India, 14–16 October 2020; pp. 1–5.
54. Xu, H.; Wang, S.; Cai, X.; Zhang, W.; Liang, X.; Li, Z. CurveLane-NAS: Unifying lane-sensitive architecture search and adaptive point blending. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 689–704.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.