LSV-LP: Large-Scale Video-Based License Plate Detection and Recognition

Qi Wang, *Senior Member, IEEE,* Xiaocheng Lu, *Student Member, IEEE,*
Cong Zhang, *Student Member, IEEE,* Yuan Yuan, *Senior Member, IEEE,* Xuelong Li, *Fellow, IEEE*

Abstract—In the past few decades, license plate detection and recognition (LPDR) systems have made great strides relying on Convolutional Neural Networks (CNN). However, these methods are evaluated on small and non-representative datasets that perform poorly in complex natural scenes. Besides, most of existing license plate datasets are based on a single image, while the information source in the actual application of license plates is frequently based on video. The mainstream algorithms also ignore the dynamic clue between consecutive frames in the video, which makes the LPDR system have a lot of room for improvement. In order to solve these problems, this paper constructs a large-scale video-based license plate dataset named LSV-LP, which consists of 1,402 videos, 401,347 frames and 364,607 annotated license plates. Compared with other data sets, LSV-LP has stronger diversity, and at the same time, it has multiple sources due to different collection methods. There may be multiple license plates in a frame, which is more in line with complex natural scenes. Based on the proposed dataset, we further design a new framework that explores the information between adjacent frames, called MFLPR-Net. In addition to these, we release the annotation tools for license plates or vehicles in videos. By evaluating the performance of MFLPR-Net and some mainstream methods, it is proved that the proposed model is superior to other LPDR systems. In order to be more intuitive, we put some samples on Google Drive. The whole dataset is available at https://github.com/Forest-art/LSV-LP.

Index Terms—Artificial intelligence, computer vision, license plate detection, license plate recognition, convolutional neural network, dataset

1 Introduction

License Plate Detection and Recognition (LPDR) is a topic of great research significance. It is an essential branch of Intelligent Transportation Systems (ITS) and computer vision. The accurate and efficient LPDR system can be widely used in various monitoring scenarios, such as traffic flow regulation, parking fee management and private spaces access. Due to the importance of LPDR in ITS, many researchers have made great improvements in this field [1], [2], [3], [4], [5], [6], [7], [8].

Benefiting from the rapid development and wide application of deep learning, current LPDR systems [9], [10], [11] are mostly based on Convolutional Neural Networks (CNN). The CNN-based system is generally divided into several sub-modules, such as vehicle detection, license plate (LP) detection, LP character segmentation and LP character recognition. These sub-modules perform their respective functions and constitute an integrated system to complete the task of LP recognition on images. Most of these studies are for LP recognition under specific tasks, and the scenes are relatively simple. For example, the AOLP [12] dataset contains only images captured from different angles or heights. It is worth noting that the real-world scenes for LPDR are complex and changeable, such as distortion, occlusion, and fog weather blur. In order to alleviate the problem of insufficient data, Xu *et al.* [2] provide a large-scale dataset containing a variety of scenarios with more than 250,000 unique images. However, in practical applications such as video surveillance and traffic management, the LPDR system relies not on a single image but on multiple frames. In [9], [13], two multi-frame datasets called UFPR-ALPR and SSIG-SegPlate are proposed which can be exploited for LPDR in videos. Nevertheless, both of these datasets contain 150 videos, each of which averages less than 30 frames, and the background in the videos is relatively monotonous. If the training processing is carried out on such datasets, it is difficult to generalize this model, and the number of different LPs is small, leading to that the system should be based on character segmentation or fine-tuning.

Considering the problems mentioned above, we propose a large-scale LP dataset named **LSV-LP**, which covers LPs captured from videos in various provinces of China. The background includes complex weather environments, a variety of time periods and different shooting scenes such as parking lots and freeways. In addition to the differences in the image background, there are also great differences in the way they are shot. The dataset contains video footage taken by phone or camera, which leads to differences in the resolution of each frame. The LP in each video sample has different inclinations, degrees of ambiguity and light environment conditions, which is more in line with the com-

- *The authors are with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P.R. China, and they are also with the Key Laboratory of Intelligent Interaction and Applications (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, P. R. China. Emails: {crabwq, xiaochenglu1997, nwpuzhangcong, y.yuan1.ieee}@gmail.com, li@nwpu.edu.cn.*

Manuscript received 27 Sept. 2021; revised 21 Jan. 2022; accepted 13 Feb. 2022. Date of publication 0.0000; date of current version 0.0000.
This work was supported by the National Natural Science Foundation of China under Grant U21B2041, 61871470, and 61825603.
(Corresponding author: Xuelong Li.)
Recommended for acceptance by C. Fermuller.
Digital Object Identifier no.10.1109/TPAMI.2022.3153691

TABLE 1
A comparison of publicly available datasets for LPDR and our proposed dataset LSV-LP. Var denotes varations. From this table, the advantages of LSV-LP in complex scenarios can be demonstrated.

	EnglishLP	CLPD	AOLP	ReID	SSIG-ALPR	SSIG-SegPlate	UFPR-ALPR	CCPD	LSV-LP
Year	2003	2020	2012	2017	2018	2015	2018	2018	2020
Number of images	509	1200	2049	76k	6660	2000	4500	301k	400k
Var in distance	✗	✓	✗	✗	✓	✓	✓	✓	✓
Var in tilt degrees	✗	✓	✓	✗	✗	✗	✗	✓	✓
Var in blur	✗	✓	✓	✓	✗	✗	✗	✓	✓
Var in illumination	✗	✗	✓	✗	✗	✗	✗	✓	✓
Var in weather	✗	✗	✓	✗	✗	✗	✗	✓	✓
Var in time	✗	✗	✗	✗	✗	✗	✗	✓	✓
Var in resolution	✗	✗	✗	✗	✗	✗	✗	✗	✓
Video based	✗	✗	✗	✓	✗	✓	✓	✗	✓
Car Annotations	✗	✗	✗	✗	✗	✗	✗	✗	✓
Vertices Annotations	✗	✗	✗	✗	✗	✗	✗	✓	✓

plex background of LPDR systems in practical applications. Not only is it diverse, but it is also massive, with over 400k video frames in total. Diversity and large volume make LSV-LP a better dataset for LPDR systems because it contains complex backgrounds and video features that can be used in combination. It is also important to note that the tags in the dataset, including the location box of the vehicle, the four vertices of the LP and the LP number are all available. Based on this, the dataset can also be applied to vehicle or LP tracking, ReID and other related tasks. Due to the large volume, we put some sample data on Google Drive[1], which makes it easier to view the data composition.

Based on the dataset we proposed, a **m**ultiple **f**rames **l**icense **p**late **r**ecognition **n**etwork called MFLPR-Net is designed in this paper, which utilizes the features between adjacent frames in the video to assist detection. On the proposed LSV-LP dataset, our algorithm achieves superior results compared with other state-of-the-art (SOTA) algorithms. Exploring the information between the frames in the video can improve the detection results and accelerate the performance of the LPDR system.

The contributions of this paper are summarized as follows:

- A large-scale and multi-source video based LPDR dataset, namely LSV-LP, is proposed, which contains more than 400k video frames with various complex scenes and diverse data sources. LSV-LP is closer to the actual scene and larger in volume, which brings more robust benefits.
- A tool for labeling vehicle information in videos is released, and multiple people can collaborate to develop labeling. The labeling information includes vehicle positioning frame, LP vertex and LP number, which brings more information that can be mined for LPDR systems.
- A novel framework based on video datasets is proposed, which can use inter-frame features to improve the recognition effect of each frame under the premise of satisfying speed. Multi-scale feature fusion is applied in the detection and recognition stages. At the same time, this pipeline combines the optical flow network to propagate features, which reduces the pressure of feature extraction in each frame.
- The experiments of various SOTA algorithms on LSV-LP verify the necessity of this dataset and the effectiveness of the proposed algorithm.

The rest of the paper consists of the following parts. In Section 2, we summarize the existing LPDR datasets and algorithms. Section 3 specifically describes the proposed dataset. Our algorithm is introduced in Section 4 and the experimental results are shown in Section 5. In the end, we discuss the conclusion and outlook.

2 Related Works

In this section, two aspects are mainly concerned: the publicly available LP datasets and the existing LPDR systems or algorithms.

2.1 Datasets for LPDR

Table 1 compares the existing datasets in detail from multiple aspects, and we specifically describe the relevant datasets and off-the-shelf algorithms in the following sections. Most of the current datasets are developed for LP identification tasks in their own countries, collected respectively from parking lots, expressway tollbooths and traffic monitoring systems. The background of these images is usually well-lit and the LPs are tilted at no more than 20° within the image. Moreover, since each country has its own LP rules [14], [15], [16], if dividing them according to national rules, there will be many kinds. In this section, we divide the existing mainstream datasets into two categories by images or multi-frame videos: image-based datasets and video-based datasets.

2.1.1 *Image-based Datasets*

AOLP [12]: Gee-Sern Hsu *et al.* proposed a Taiwanese LP dataset that was divided into three subsets: Access Control (AC), Law Enforcement (RP) and Road Patrol (LE). AC referred to the situation where a vehicle passed through a

1. https://drive.google.com/file/d/1udqRddpJZMpTdHHQdwZRll6vaYALUiql/view?usp=sharing. The complete dataset is available at https://github.com/Forest-art/LSV-LP.

fixed passage by slowing down or stopping completely. LE referred to violations of traffic laws captured by roadside cameras, while RP referred to images of vehicles taken from arbitrary viewpoints and distances with cameras installed or held on patrol vehicles. The three subsets of AC, LE, and RP contained 681, 757, and 611 images, respectively, for a total of 2049 images. This dataset was diversified in four aspects: horizontal, vertical, shooting roll, and shooting distance, which lacked more complex backgrounds such as dark light, blurring, and terrible weather. In 2017, an additional version of the dataset called **AOLPE** was proposed in [17], which contained 4,200 images, but still lacked such harsh conditions as blurriness.

CCPD [2]: Xu and Yang *et al.* constructed a huge LP dataset for China Mainland, containing more than 290k images under a variety of conditions. This dataset consisted of nine subsets, Base, FN, DB, Rotate, Tilt, Weather, Challenge, Blur, and NP, and the backgrounds covered various complicated situations such as blur, occlusion, bad weather, distance, and tilt. Although there are many types of LPs in CCPD, with the development of new energy technology, green and energy-saving vehicle LPs are gradually integrated into our life. In 2020, the authors extended CCPD and proposed a CCPD-green subset for new energy green LPs, which contains more than 11k samples. Together with the previously proposed CCPD which contains more than 290k images, this dataset totals 301k images. The data scale of CCPD was large enough to achieve better LP recognition effect in static scenes. However, some problems such as motion blur required multi-frame information for fusion recognition, which could not be well utilized in the dynamic and complex background in the real world.

EnglishLP [18]: SrebrŠ proposed an European LP dataset named EnglishLP, which contained 509 images. This dataset was all taken from the rear of the car and didn't have the universality of the LP in complex scenes. At the same time, it had no uniform markings of LP location box and LP number. Therefore, it is rarely used.

SSIG-ALPR [10]: Gonçalves *et al.* annotated and published a Brazilian LP dataset for the ALPR system, called SSIG-ALPR. This dataset contains 6,660 images with 8,683 LPs from 815 different on-track vehicles. However, 3,368 LPs have no text annotations because their resolution is very low and their characters cannot be determined intuitively.

CLPD [19]: Zhang *et al.* proposed a real LP dateset, which contains 1200 images of all provinces in mainland China, including different vehicle types. The images in CLPD dataset are all based on the real environment, which covers a variety of shooting conditions and area codes. The authors trained their own model on CCPD dataset and validated it on this dataset.

2.1.2 Video-based Datasets

ReID [20]: Špaňhel *et al.* developed a large video-based dataset that contained 14,360 tracks and more than 170k images. This dataset was able to extract more than 76k LPs and annotations, all collected from surveillance cameras on highway. The images in ReID lacked tilt angle samples, occlusion samples, and could not be used in more complex LPDR systems.

Fig. 1. The display of the proposed LSV-LP dataset with annotations. The first column of the video sequence is *move vs. static*, the second column is *move vs. move*, and the last column is the type of *static vs. move*.

SSIG-SegPlate [13]: In addition to the image-based dataset SSIG-ALPR [10], Gonçalves *et al.* also proposed a public dataset of Brazilian LPs, which contained less than 800 training examples. This dataset had 40 videos for training, 21 for validation and 40 for testing, with each video no more than 30 frames. However, this dataset had several constraints, including the use of a fixed camera, the absence of a double-character motorcycle LPs, and a single background.

UFPR-ALPR [9]: In order to effectively remedy the deficiency of SSIG-SegPlate, Laroca *et al.* proposed the UFPR-ALPR dataset. Compared with [13], it had a larger scale, specifically including 60 training videos, 30 validation videos and 60 testing videos, while each video regularly contained 30 frames. Moreover, it complemented many motorcycle LP samples, which were even more challenging. Still, each video in this dataset contained only one car and the background is simple. Meanwhile, there were only 150 different LP samples, which highly depended on character segmentation to effectively complete LP recognition task.

2.2 Methods for LPDR

A complete LPDR system aims to input an image, locate the LP position from the image, and fully recognize the LP characters within the positioning area. Therefore, an LPDR system can be divided into at least two stages, LP detection and LP recognition, even for the model of end-to-end training. In the following, we will introduce the mainstream LP detection and LP recognition algorithm respectively.

2.2.1 LP Detection Algorithms

Existing LP detection algorithms can be coarsely divided into traditional methods and deep learning based methods.

Most of the traditional LP detection methods use the edge, color or texture features of the LP region. Yu *et al.* [21] utilized wavelet transformation and EMD analysis to locate LPs in images. Saha *et al.* [22] proposed a multi-stage method that analyzed vertical edge gradients to select the right area. In [23], a classifier based on cascade AdaBoost and a voting module are designed to vote for the candidate regions of LPs. Lee *et al.* [24] proposed a method using local structure patterns to locate the LPs in images. Some other researchers [25], [26] applied HSI color model and a color checking module to detect the candidate regions containing LPs.

At present, the LP detection algorithms based on deep learning take the LP as an object and use some mainstream object detection models such as YOLO series [27], [28] and SSD [29] to locate the regions of LPs. Hse *et al.* [17] combined YOLO and YOLOv2 [30] to design a larger network that computes possibilities for both LP and background. Since the series of models based on YOLO is not suitable for the detection of small objects, this algorithm needs to be improved for images taken from a longer distance. In [3], two YOLO based networks were constructed, and one was the attention module focusing on the region where LPs were located, while the other was used to locate the LPs with rotation. Since this algorithm was also based on YOLO, it had the same problem as [3]. Besides, it was proposed for the presence of rotating LPs, while it was difficult to demonstrate its performance for more complex deformations. [31] and [32] utilized synthetic datasets for training, and migrated models to real data for text localization. However, as it was a composite data, there were still many limitations in practical application scenarios. The system proposed in [33] is based on Mask-RCNN [34] architecture and classifies the extracted regions into "LP found" and "LP not found". They merged several datasets as positive and negative samples during the training process. Chen *et al.* [35] utilized a full convolutional network as a pixel-level binary classification method, which performed random target detection by fusing multi-scale and hierarchical features. Wang *et al.* [36] proposed VertexNet in the LP detection stage, which was an effective integrated block to extract the spatial characteristics of LPs. With vertex supervision information, they proposed a vertex estimation branch in VertexNet, so that the LP can be used as the input image for LP recognition for correction.

2.2.2 LP Recognition Algorithms

Different from LP detection algorithms, LP recognition algorithms could be classified into two categories, segmentation-based methods and segmentation-free methods.

Segmentation-based approaches usually split the characters first and then recognize the segmented regions. As for the segmentation stage, Zhuang *et al.* [37] introduced the Connected Components Analysis (CCA). Then, Inception-V3 and AlexNet were adopted as the character classification and character counting models. Montazzolli and Jung [38] proposed a CNN-based algorithm for character segmentation and recognition. After segmentation, the recognized LP characters were cropped out and fed into the character classification model. Laroca *et al.* [9] utilized two different depth of networks for the character recognition.

For segmentation-free LP recognition, CNN was generally used for feature extraction while LSTM and other networks were applied to output the complete LP numbers. In order to relieve the pressure of the recognition network, Dong *et al.* [39] added a spatial-temporal sampling network for deformation correction of LP area before recognition. Besides, Svoboda *et al.* [40] utilized a CNN based approach for LP deblurring to effectively alleviate the problem of motion blur. The authors in [41] proposed a simple yet effective intensity- and gradient-based L0 regularization prior for text image deblurring. The proposed image prior was based on the unique properties of text images, and they developed an efficient optimization algorithm to generate reliable intermediate results for kernel estimation. The proposed algorithm did not require any heuristic edge selection methods that were crucial to state-of-the-art edge-based deblurring methods. In addition to the methods described above, there are some text recognition methods that can be applied to this. Shi *et al.* [42] introduced the Connectionist Temporal Classification (CTC) loss to the text recognition. Besides, Wojna *et al.* [43] and Luo *et al.* [44] utilized the attention mechanism to the model to enhance the result of the recognition. In addition to these, Wang *et al.* [45] also designed a decoupled encoder to process text recognition to improve robustness. Zhang *et al.* [19] proposed a powerful LP recognition framework. which consisted of a customized CycleGAN model for generating LP images and a carefully designed image-to-sequence network for LP recognition. In [35], an extensive learning system of stacked autoencoders with mapped feature nodes was proposed, and two structures were explored to recognize letters and numbers respectively. Wang *et al.* [36] introduced a horizontal encoding technique for feature extraction from left to right, and proposed a weight sharing classifier for character recognition.

3 LSV-LP Dataset

This section describes the established LSV-LP dataset from three aspects, namely: data collection and specification, annotation tool and dataset characteristics. Data collection sources, classification methods, and data specifications are introduced in the data collection and specifications section. The annotation tool specifies the design of the tool and the process of information annotation. Finally, specific analysis and feature summary are made in the data feature part. Annotation tool and dataset samples have been released, while the complete dataset and annotations will be open soon after the paper is accepted.

3.1 Data Collection and Specification

Data Collection. Our data collection mainly comes from driving recorders, street camera shooting and mobile phone shooting, which are all taken and organized by ourselves. Benefiting from different capturing methods, the sizes of the videos in LSV-LP are various, and the frame number in each video is consistent. The smallest resolution is 368×640 while the largest is 1920×1080. The shooting locations include highways, streets, parking lots and other scenes, covering 27 provinces of China mainland. In addition, the shooting time and background are also diversified, including complex situations under various conditions from morning to evening, from sunny to snowy. In the proposed dataset, each long video is carefully divided into video clips, which ensures that the total frames in each video clip is limited to 300 for the convenience of algorithm training and evaluation. It is worth noting that not every frame in the video contains LP, and these unlabeled frames will be regarded as negative samples in the dataset.

Data Specification. There is a big difference between the resolution of the video taken by the mobile phone and the video taken by the driving recorder, but there is not much

Fig. 2. Typical examples of the proposed LSV-LP dataset. The first line of the video sequence is *move vs. static*, the second line is *static vs. move*, and the third line is *move vs. move*. The actual frame interval between two adjacent frames in the figure is 50.

difference in other aspects. Considering that different shooting methods only bring about differences in resolution, we do not classify the videos in the dataset strictly according to the shooting method but classify them according to whether the photographer and the captured vehicles are stationary or moving. It is divided into three categories, namely: *static vs. move*, *move vs. static*, and *move vs. move*. Taking *static vs. move* as an example, it shows that the photographer is still while the vehicle is moving. Fig. 2 shows typical examples of the three types in LSV-LP.

In this way, the design of the algorithm would be more targeted for practical applications. Next, we specifically discuss the data characteristics of each category. Relative to the background, the vehicles in *static vs. move* are moving, and it is easier to locate the vehicles by using the contextual information in the video. Since this type of videos are usually shot at a road intersection, each frame contains a lot of vehicles, and there are complicated situations such as LP occlusion and high LP density. In contrast, *move vs. static* mostly comes from holding a mobile phone to take pictures of parked cars on the roadside or parking lot. Due to the instability of the shooting phone, there are large differences between adjacent frames. The *move vs. move* is generally collected in the driving recorder, which covers the various characteristics of *move vs. static* and *static vs. move*. The details can be seen in Table 2.

3.2 Data Annotation

Annotation Tool. In order to facilitate the labeling of LPs in videos, we develop and release an efficient online labeling tool[2] based on HTML5 + Javascript + Python. This tool supports marking the vehicle location, LP location and LP number in the video. It is necessary to explain that the vehicle is marked as a rectangular box, and the LP is marked as an arbitrary quadrilateral, so that the tilt or distortion of the LP can be used in the algorithm. In order to make labeling more efficient, this tool can label non-continuous frames and perform linear interpolation on the intermediate frames. The marked video demo can be seen in the tool website.

Vehicle Annotation. There are multiple vehicles in a video and the LPs have multiple resolutions. A unified standard under which the LP needs to be marked should be designed. In LSV-LP, we label vehicles with a LP resolution of more than 20×8 pixels in the video. Even if the vehicle can be seen clearly, its LP resolution is too low and will not be marked. Since the LP is already invisible at this resolution, the following annotations have no practical value.

LP Annotation. Different from the rectangular frame labeling of vehicles, the LPs have rotation and distortion in videos, and inaccurate labeling will have greater impact on the recognition of the LPs. In order to enhance the accuracy of labeling, we label the four vertices of LPs. As mentioned above, when the resolution of the LP is lower than 20×8, the label is abandoned.

However, there is a situation where the resolution is higher than 20×8 and the LPs cannot be seen clearly in all frames in the video. For this case, we use # to replace these unclear LP characters. In addition, when the characters of the LP are individually occluded and affect the recognition, the labels of the vehicle and the LP are discarded.

3.3 Data Characteristic

LSV-LP dataset consists of $1,402$ videos totally, with $401,347$ frames and $364,607$ annotated LP instances. As mentioned above, compared with other existing LP datasets, it is the largest from the perspective of three different levels, *i.e.* videos, images, and instances. It is worth noting that some Chinese LP datasets are introduced publicly or privately, but most of them are collected from one province or region. For example, CCPD [2] only contains LPs from Anhui Province, which means that the first Chinese character of all LPs is "WAN". However, compared to other English letters and numbers, Chinese character recognition is a more challenging task, and its accuracy limits the end-to-end performance of LPDR systems, which deserves more research

TABLE 2
The number of videos, frames, LPs and the average resolution of the subsets in the LSV-LP.

Subset	Number of videos	Number of frames	Number of LPs	Avg. Resolution (W×H)
move vs. move	504	145,706	157,226	1896×1066
move vs. static	600	167,012	69,353	722×1220
static vs. move	298	88,629	13,8028	1870×1052
Total	1402	401,347	364,607	1402×1127

2. https://github.com/Elin24/license_labeler

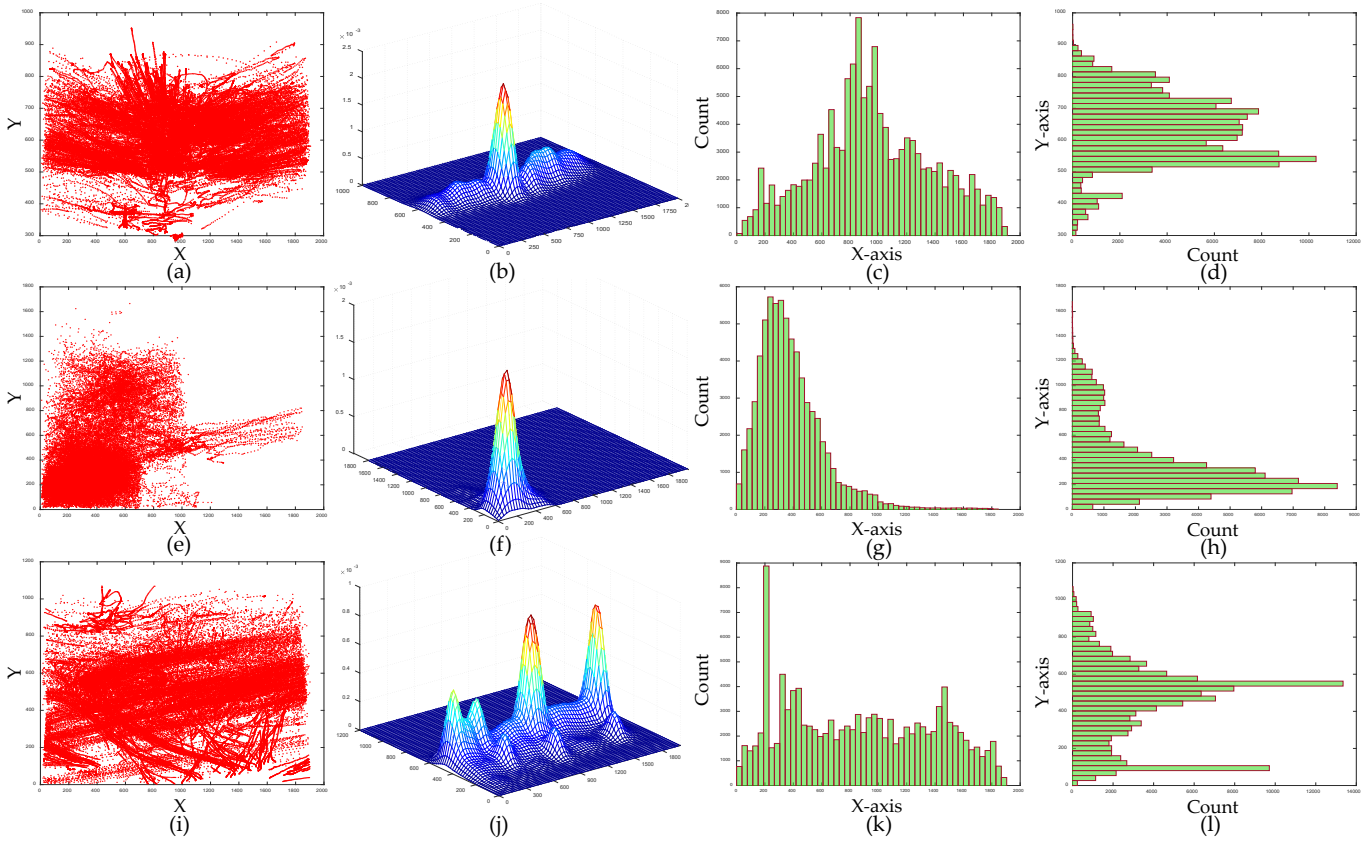


Fig. 3. Differences in the position distribution of LP in the three subsets of images. (a)-(d) for the *move vs. move* subset, (e)-(h) for the *move vs. static* subset, and (i)-(l) for the *static vs. move* subset. (a), (e), and (i) represent the statistical maps of the LP distribution. (b), (f), and (j) mean the probability density maps obtained by 2-dimensional Parzen-window estimation with Gaussian kernel. (c), (g), and (k) are the density maps on X-aixs. (d), (h), and (l) are the density maps on Y-aixs.

attention. To address this issue in other Chinese LP datasets, the proposed LSV-LP dataset includes LPs collected from multiple provinces, making Chinese characters diverse. As illustrated in Fig. 4, it covers almost all the Chinese LP characters for different provinces, which enriches the character diversity especially for real-world applications. In addition to data volume and character diversity, LSV-LP has five more advantages compared to previous LP datasets:

1) **Distributional Diversity.** LSV-LP dataset has the complicated spatial distribution of LPs. Most previous datasets only contain simple data distributions since LPs always appear at specific positions in the images, such as the center caused by manual capture. However, in some complex scenarios like ITS, LPs may show at any random position without prior knowledge. Considering practical application requirements, the proposed LSV-LP dataset includes videos with extensive and complex data distribution. As shown in Fig. 3, there are some distributional differences in the three subsets. The first column is a scatter plot of the LP location distribution. It can be seen that the vehicle trajectories are chaotic due to unpredictable traffic conditions in the real world, which guarantees distributional diversity against restricted arrangement. The second column is the LP distribution density map at each location. It can be seen that although the overall spatial distribution of LP is chaotic, its distribution density also has a certain concentration some locations. The third and fourth columns are the LP position distribution from the x-axis and y-axis, which mainly subdivide the density map.
2) **Negative Samples.** Different from existing LP datasets which only involve positive samples, LSV-LP proposes considerable negative samples without LPs. It has been proven that unbalanced positive and negative training samples can affect model performance. For LP detection, there are many interfering objects similar to LPs in terms of texture features in real-world driving scenarios. These negative samples contain traffic signs, street view texts, and other scenes, which significantly reduce detection precision. Most previous algorithms solve this problem by data augmentation. For example, in the previous object detection datasets, the regions without objects can be cropped and taken as the negative samples. Nevertheless, this strategy is resource-consuming and limited by the background similarity. LSV-LP dataset introduces redundant temporal frames that present no LP features as negative samples, which breaks through some limitations. In this way, LSV-LP can be exploited directly for training and evaluation due to balanced sample distribution and sufficient data. Moreover, these samples effectively improve

the generalization of LPDR models for real-world applications.

3) **Temporal Complexity.** Many previous datasets focus on simple and specific scenarios. For instance, in the widely used AOLP dataset [12], there are only vehicle bodies and LPs in each image while CCPD [2] introduces LP images in relatively complex scenarios involving uneven illuminations, rotation, and blur. However, their complex patterns are still limited. More importantly, as image-based LP datasets, they lack temporal complexity. For spatial complexity at the image level, there are complicated background objects, extreme illuminations, and perspective distortions in the proposed LSV-LP dataset. Meanwhile, thanks to the collected videos in real-world driving or traffic scenarios, it also involves temporal complexity. The vehicles and LPs may move with fast speed and significantly changeable trajectory, leading to complex temporal distribution and motion blur. As depicted in Table 1, all crucial types of variance are included in LSV-LP dataset.
4) **Various Resolutions.** Different from most existing datasets that usually contain images of specific resolutions like 1920×1080, LSV-LP dataset involves various video/image resolutions. It collects both high- and low-resolution scenes, which is entailed for LPDR in extremely complex scenarios. As illustrated in Table 2, the average resolution of LSV-LP is 1402×1127. Three different subsets have diverse resolutions while the resolution varies in each subset. This characteristic makes the proposed dataset more suitable for real-world LPDR applications in complex and unconstrained scenarios since it retains the diversity of video capture devices, which further implies new challenges and requirements for future LPDR algorithms.
5) **Large Appearance Variation.** The LP appearance changes significantly due to various shooting angles and camera-LP distances. As shown in Fig. 5, the pixel spans of the intances vary from 0 to 17.3%, which means the large variation in the ratio of LP regions to the whole images. The smallest LP only occupies 18 pixels while the largest one covers more than 357934 pixels. Since the data is concentrated in the previous section, for better visulization, we separately display the number of LPs with an area of 0-250k pixels, as shown in Fig. 6. It shows the distribution of the number of pixels occupied by LPs. Since the proportion of LPs in Fig. 5 is mainly concentrated in the part before 0.025, we only zoom in and observe the distribution of the number of pixels in the first 250k. As a whole, it can be seen that the LPs with the most pixels are concentrated around 10k, which is directly related to the way the data is captured. *Static vs. move* and *move vs. move* LPs are generally smaller, while LPs with larger pixel counts are mainly concentrated in *move vs. static*. In addition to the difference in pixel distribution, we also count the difference in the horizontal and vertical inclination of LPs. Table 3 compares the horizontal and vertical tilt degrees of AOLP, CCPD, CLPD, and LSV-LP. Our data achieves 70 degrees of horizontal and vertical inclination, outperforming the other three datasets in both dimensions. It can be seen that the LP situation in LSV-LP is more complicated and challenging. This variation may bring more challenges and inspirations to LP detection and recognition tasks. For instance, the detection performance of small-size LPs can be specifically studied on LSV-LP dataset.

In summary, LSV-LP is one of the largest and most challenging LP datasets at present.

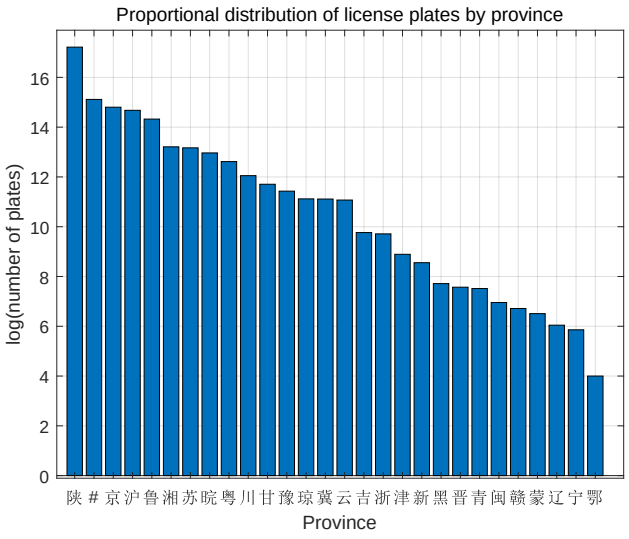


Fig. 4. The display of the proportional distribution curve of LP areas to the examined frames. The abscissa represents the proportion, and the ordinate represents the total number of LPs under the current proportion.

4 METHODS

In this section, we design a network called **MFLPR-Net** (**M**ultiple **F**rames **L**icense **P**late **R**ecognition **Net**work) that explores the features of the neighboring frames for video based scenes.

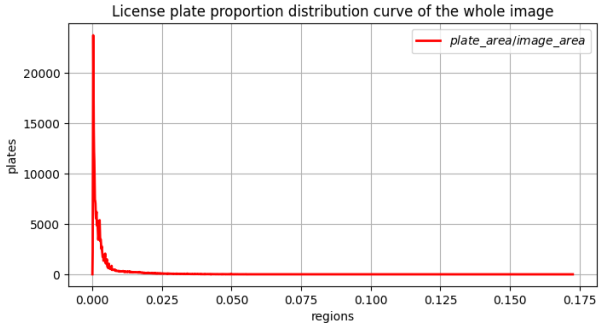


Fig. 5. The display of the proportional distribution curve of LP areas to the examined frames. The abscissa represents the proportion, and the ordinate represents the total number of LPs under the current proportion.

TABLE 3
Comparison results of tilt angles of AOLP, CCPD, CLPD and LSV-LP.

	AOLP	CCPD	CLPD	LSV-LP
Horizontal Tilt Degree	0°∼70°	0°∼50°	0°∼45°	0°∼70°
Vertical Tilt Degree	0°∼60°	0°∼45°	0°∼60°	0°∼70°

4.1 Model Preview

Our proposed dataset LSV-LP has more inter-frame information available than other LP datasets. At present, most LPDR systems detect and recognize LPs in images instead of videos, and in-video detection algorithms such as FGFA detect large objects, which cannot be well applied to our dataset. The optical flow module in FGFA [46] will bring a lot of time consumption to the LPDR system. Besides, the optical flow module in DFF [47] increases the time consumption of the entire model but reduces the accuracy. Based on this, it is necessary to propose a novel LPDR system for video scenes.

According to the investigation of the existing LPDR system in related work, most of the algorithms add super-resolution, deformation correction and vehicle detection modules to the basic module of LPDR to improve the performance. This may enhance the system in some aspects, but it will greatly increase the complexity of the system, resulting in low real-time performance. We hope that our model is as concise as possible, and can use the temporal information under the premise of ensuring accuracy without increasing time cost.

Compared with the traditional object, LP has a very small area in each frame. This is a big challenge for detection and recognition, and feature fusion between frames will have a great impact on detection. Considering these issues comprehensively, we design a multi-scale feature fusion structure in the detection stage, introduce affine transformation in the recognition stage, and concate the features of adjacent frames with the examined frame features in the feature propagation stage instead of fusing them.The optical flow algorithm is used to combine the features of adjacent frames. Of course, there are many other video feature propagation methods such as LSTM and tracking algorithms. The optical flow algorithm has two advantages:

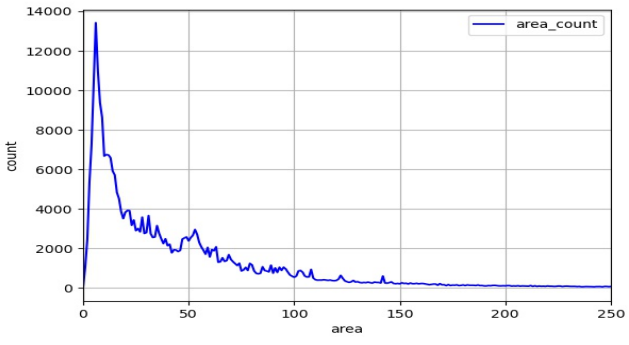


Fig. 6. The display of the distribution curve of LP areas. The abscissa represents the area, and the ordinate represents the total number of LPs. The abscissa is the result of dividing the number of pixels by 1000.

1) It does not need to rely on the previous multi-frame information like LSTM and other networks; 2) It does not need to separate detection and tracking like tracking-based algorithms.

4.2 Model Design

The original intention of our model design is to make the recognition effect as good as possible when the detection and recognition module is lightweight and the reference frame information can be used well. In the detection and recognition stage, a multi-scale U-shaped network is used to adapt to changes in image resolution and LP size. At the same time, the optical flow network can quickly calculate the feature difference between frames and propagate it to the examined frame. Based on these, we design MFLPR-Net which can be simply divided into three modules in Fig. 7. For the first part, it is a module that combines the U-shaped network of different scale feature maps for LP detection, named as 'Detection Module'. It takes advantage of networks such as VGG16 or Resnet as the backbone to extract four features of different sizes at different stages for subsequent upsampling. For the second part, it uses video context information, exploiting the optical flow module to propagate features of adjacent frames to enhance the detection results of the examined frame. We name this module 'Optical Flow Module'. The final module is the 'Recognition Module', which uses affine transformation operation to crop and correct the area where the LP is located and uses the LSTM and decoupled text decoder to directly identify it.

Detection Module In our model, we utilize the VGG16 network as backbone to extract features from the input image. After pooling-2 to pooling-5, four feature maps with different sizes of $1/32$, $1/16$, $1/8$ and $1/4$ of the input image are extracted for subsequent up-sampling operation and feature fusion. The detection and recognition of each image are based on the feature extracted by the feature extraction module.

Over the backbone we get four features of different sizes, each of which is then amplified by the upsampling operation and the concatenation operation to create new feature maps. The manipulation of these features is shown in Fig. 8.

In the feature concatenation stage, we gradually merge them by:

$$g_i = \begin{cases} U\left(h_i\right) & if \quad i \leq 3 \\ C_3\left(h_i\right) & if \quad i = 4, \end{cases} \tag{1}$$

$$h_i = \begin{cases} f_i & if \quad i = 1 \\ C_3\left(C_1\left(\left[g_{i-1}; f_i\right]\right)\right) & otherwise, \end{cases} \tag{2}$$

where f_i is the feature extracted by the backbone, h_i is the merged feature map and g_i is the intermediate feature. In addition to these, $U\left(h_i\right)$ means bilinear upsampling of

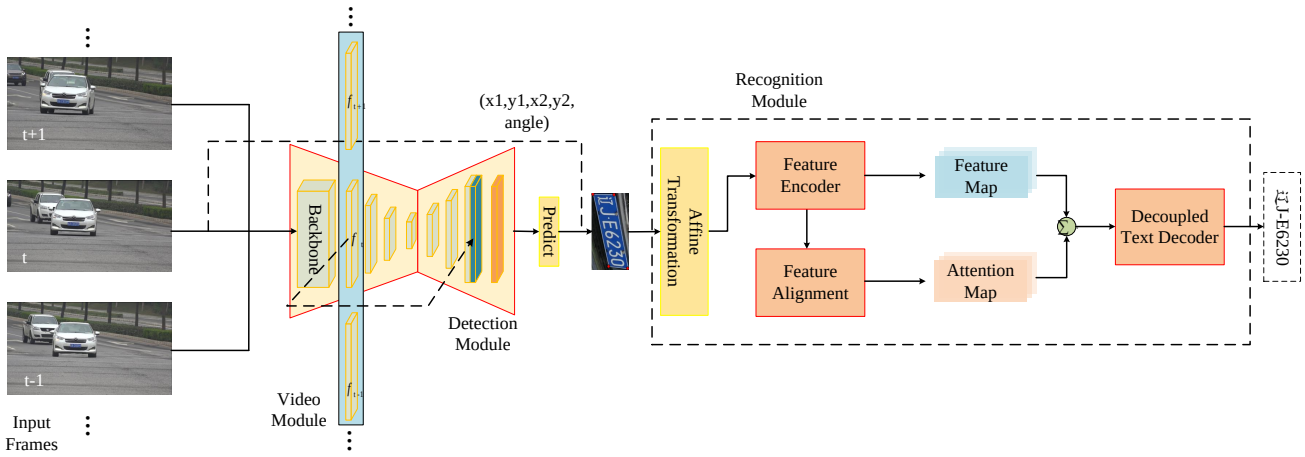


Fig. 7. The overall structure of the proposed MFLPR-Net. It consists of Detection Module, Optical Flow Module and Recognition Module. The input of the entire network is the continuous multi-frame images. These frames generate continuous feature f_i through backbone and four convolution layers. The optical flow module is responsible for assigning weights to these features and aggregating them. f_i is updated as new features and sent to the latter part of the detection module and the LP position is output. The recognition module corrects the rotating region of the LP according to the detected coordinates, then extracts the features, and finally outputs the LP characters in frame i.

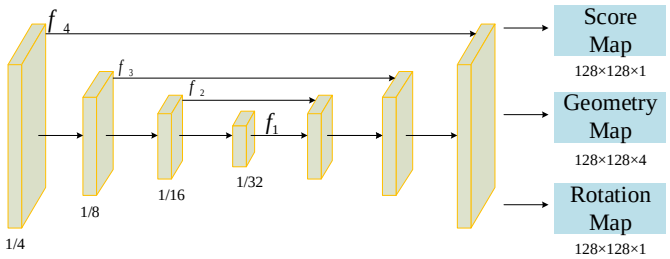


Fig. 8. The overall structure of the detection module without optical flow module is a U-shaped network structure. First, the size is reduced by pooling and then the size is expanded by upsampling.

h_i, C_1 and C_3 represent the convolution of 1×1 and 3×3 respectively, and the operator $[g_{i-1}; f_i]$ represents the concatenation along g_{i-1} and f_i. In each concatenation stage, the feature map extracted from backbone is first fed into an upsampling layer to double its size and then concatenated with the examined frame feature map. The new features obtained after the concatenation go through two convolutional layers for the next stage of the concatenation. Following the final concatenation stage, a convolution layer of 3×3 produces the feature map for the output layer.

The last output convolution layer contains three $conv_{1\times1}$ operations to output 1 channel of score map F_s and the 5 channels of geometry map F_g. The 5 channels of F_g are composed of four positioning channels and one rotation angle channel. Based on this, for each pixel which has positive score, we calculate its distances to 4 boundaries of the quadrangle. The loss function for classification can be expressed as follows:

$$L_{cls} = \frac{1}{|F_s|} \sum_{x \in F_s} CE\left(p_x, \breve{p}_x\right) \tag{3}$$

$$= \frac{1}{|F_s|} \sum_{x \in F_s} \left(-\breve{p}_x \log p_x - \left(1 - \breve{p}_x\right) \log\left(1 - p_x\right)\right), \tag{4}$$

where $CE\left(p_x, \breve{p}_x\right)$ represents the cross entropy loss between the prediction of the score map and the binary label. For the loss of regression stage, we utilize the IoU loss, which is formulated as follows:

$$L_{reg} = \frac{1}{|F_g|} \sum_{x \in F_g} I\text{oU}\left(p_x, \breve{p}_x\right) \tag{5}$$

$$= \frac{1}{|F_g|} \sum_{x \in F_g} -\ln \frac{Inter\,sec\,tion\left(p_x, \breve{p}_x\right)}{Union\left(p_x, \breve{p}_x\right)}, \tag{6}$$

where $I\text{oU}\left(p_x, \breve{p}_x\right)$ denotes the loss of the intersection over union between the predicted boxes and the ground-truth. Finally, the loss of the rotation angle is formulated as follows:

$$L_\theta\left(\breve{\theta}, \theta^*\right) = 1 - cos\left(\breve{\theta} - \theta^*\right). \tag{7}$$

where $\breve{\theta}$ is the prediction to the rotation angle and θ^* denotes the ground truth. Consequently, the whole detection loss can be written as:

$$L_{\ detect} = L_{cls} + \lambda_{reg} L_{reg} + \lambda_\theta L_\theta. \tag{8}$$

The parameter λ_{reg} is set to 1 and λ_θ is set to 10 in our experiments to balance the classification loss and the regression loss. The entire detection module is based on the EAST [48] model, which is efficient enough to handle the LP detection task in a single image.

Optical Flow Module The detection module is a horizontal processing process, while the optical flow module is a vertical process interspersed with it. The feature f_t in the detection module, that is, the feature with the size of $1/4$ of the original image, is processed in the optical flow module.

We define the examined frame of the LP to be detected as F_{cur}, and its adjacent frames as F_{refi}. These frames will be extracted as features f_{cur}, f_{refi} when they pass through the size reduction stage of Detection Module. The features of adjacent frames f_{refi} are added through a weight assignment module, which can be expressed as follows:

$$w_{ref} = Sigmoid(C_1(C_1(f_{refi} + f_{ref-i}))) \quad (i \leq K), \tag{9}$$

$$f_{ref} = w_{ref} \times f_{refi} + (1 - w_{ref}) \times f_{ref-i}, \tag{10}$$

where C_1 denotes the 1×1 convolutional layer and w_{ref} represents the weight of the features f_{refi}, f_{ref-i} that are symmetric with respect to the examined frame f_{cur}. The features of adjacent frames generate new feature map f_{ref} through weight allocation, which integrates the information of adjacent frames and plays an auxiliary role in the examined frame. In addition, K represents the range of adjacent frames selected, and in our experiment, K is set to 1.

The feature extraction is a cascading process, that is, the previous features will have a chain effect on the subsequent features. Based on this, we do not aggregate f_{ref} directly to the feature f_{cur} of the current stage, but concatenate it with the feature map after the upsampling. After the connected features are dimensionalized by 1×1 convolution, 32-channel feature maps are obtained for subsequent recognition tasks.

Recognition Module After predicting the bounding boxes of the LPs, we crop the area of the boxes and utilize the recognition module to output the characters of them. It is expected that the convolution parameters of the detection stage and recognition stage will not be shared during feature extraction, which can improve the flexibility of model learning. For the cropping stage, the affine transformation is utilized to cut out the position of the plate and correct the shape of the plate. Compared to RoIPooling [49] and RoIAlign [34], the affine transformation operation utilizes bilinear interpolation to compute the cropping of the area, which is a more general operation for extracting regions of text recognition. As for this process, we compute the affine parameters via the predicted bounding boxes first. Then, the affine transformation is applied to crop the detected area and correct it to a regular shape. The first step can be formulated as:

$$\begin{pmatrix} x_i^s \\ y_i^s \\ 1 \end{pmatrix} = M^{-1} \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix}, \tag{11}$$

$$M = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \tag{12}$$

where M is the affine transformation matrix. x_i^s and y_i^s are the coordinates of the input area, while x_i^t and y_i^t are the coordinates of the transformed region. According to the four point coordinates of the input area and the four point coordinates of the output area, the M matrix can be calculated, and the entire clipped area can be multiplied by the M matrix to obtain the corrected LP area.

The insertion of deformation correction module can effectively improve the recognition of plate with irregular shape, but the accurate recognition of plate still needs to be improved. Attention in text recognition is generally used for feature alignment and text recognition. It mainly uses two parts of information: visual encoding features output by encoder and historical decoding information. The traditional attention mechanism often encounters serious alignment problems, and the coupling relationship in the decoding process inevitably leads to the accumulation and propagation of errors. Inspired by [45], we utilize the decoupled text decoder to recognize the charactes of LPs.

The whole recognition module consists of three parts: feature encoder, feature alignment and decouled text decoder. We define E as the feature encoder, which has the following form:

$$F = E(x), F \in R^{C \times H/r_h \times W/r_w}, \tag{13}$$

where H,W denote the input image x of size $H \times W$, r_h and r_w denote the height and width downsampling ratio respectively. In the experiment, we utilize a series of res-blocks to carry out feature downsampling, while the feature alignment module uses several deconvolution layers to carry out upsampling and output multiple attention maps $A = \{\alpha_1, \alpha_2, \cdots, \alpha_{maxT}\}$. $maxT$ represents the maximum number of channels, and the size of each feature map is $H/r_h \times W/r_w$. From Fig. 9, it can be seen that the whole feature coding, feature alignment module and feature extraction stage of Detection Module are similar, and they are all U-shaped. The structure of the decouled text decoder is shown in Figure 6. The method of context vector c_t calculation is as follows:

$$c_t = \sum_{x=1}^{W/r_w} \sum_{y=1}^{H/r_h} \alpha_{t,x,y} F_{x,y}. \tag{14}$$

These vectors get the output y_t of the classifier through the GRU. The formula is as follows:

$$\begin{cases} y_t = wh_t + b \\ h_t = GRU((e_{t-1}, c_t), h_{t-1}) \end{cases}, \tag{15}$$

where h_t is the hidden state of the GRU in time step t, and e_t is an embedding vector of the last decoding result y_t.

The loss function of Recognition Module is as follows:

$$L_{recog} = -\sum_{t=1}^{T} \log P(g_t|I, \varphi), \tag{16}$$

where φ and g_t represent all trainable parameters and ground-truths at step t. The whole system can be trained jointly, and the full loss can be formulated as:

$$L = L_{detect} + \lambda_{recog} L_{recog}, \tag{17}$$

where the parameter λ_{recog} is set to 1 in the method.

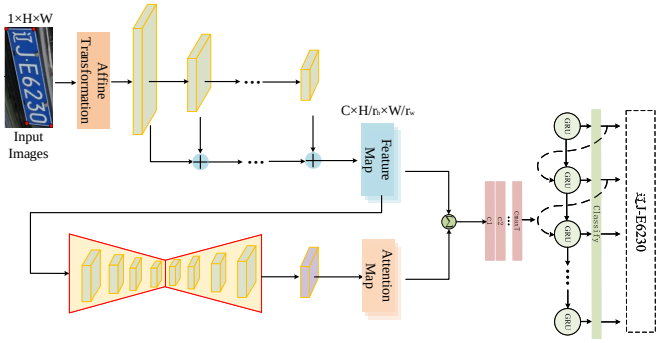


Fig. 9. The display of the overall structure of the Recognition Module in MFLPR-Net. The feature extraction module is used to obtain the feature map, and the feature alignment module is used to generate the attention diagram and redistribute the weight and then produce the output of the network.

5 Evaluations

In this section, we conduct the experiments on both LP detection and recognition performance to compare the method with other state-of-the-art approaches. And the analysis of the experimental results is also presented in this section.

5.1 Mainstream Methods for LP Detection and Recognition Comparison

5.1.1 LP Detection Methods

EfficientDet [50]: The authors propose a simple and efficient weighted bidirectional feature pyramid network BiFPN, which introduces learnable weights for learning the importance of different input features while repeatedly using bottom-up, top-up, and top-down multi-scale feature fusion. At the same time, for the problem of object detection, a hybrid scaling method is proposed, which can uniformly scale the resolution, depth and width of the network backbone, feature network, and bounding box/category prediction network. They found that the recently proposed EfficientNet is more efficient than the previously commonly used backbone structures such as ResNets, ResNetXt and AmoebaNet. Therefore, the authors combine the proposed BiFPN and hybrid scaling with the EfficientNet structure and name it EfficientDet.

YOLOv3 [51]: YOLOv3 utilizes multi-scale features for object detection, and it exploits the logistic to replace softmax in classification. It improves the prediction accuracy while maintaining the speed advantage, and especially strengthens the detection ability of small objects. Although there are already YOLOv4 and v5, they are not released in published papers, so we do not take these as comparative experiments.

Faster-RCNN [49]: In this work, a Region Proposal Network (RPN) is proposed which generates high-quality region proposals for detection. In general, it can detect small objects but relatively sacrifices running time.

SSD512 [29]: SSD is a single-stage detector, which is more accurate and faster than the previous algorithm YOLO, and does not use RPN and Pooling operations. It utilizes a small convolution filter to be applied in different feature map layers, achieving better detection effect in a smaller input image compared with Faster-RCNN.

DFF [47]: DFF (Deep Feature Flow Network) utilizes the optical flow network to propagate the features between video frames, which has the advantage of running time in the task of video object detection. In addition, the authors also distinguish between key frames and non-key frames. Feature extraction and detection are performed on key frames, while non-key frames propagate key frame features to non-key frames through optical flow for detection.

FGFA [46]: Compared with DFF, FGFA (Flow Guided Feature Aggregation Network) does not distinguish between key frames and non-key frames. It aggregates the features of adjacent frames into the examined frame through an optical flow network for detection. The features of each frame are integrated with the information of nearby frames, which has high detection accuracy in video object detection.

LPDNet [52]: End-to-end trainable network for degraded LP detection. LPDNet is proposed to estimate the local region around the LP via vehicle-plate relation mining, reducing the search area for LPs. Besides, compared with other object detection models, LPDNet localizes the quadrilateral bounding box of the oblique LP by regressing the four corners of the LP.

5.1.2 LP Recognition Methods

CRNN [42]: It is believed that text recognition is a method of sequence prediction, so CRNN utilizes Recurrent Neural Network (RNN) for sequence prediction. After extracting the features of the image through CNN, the sequence is predicted by RNN, and finally the final result is obtained through a Connectionist Temporal Classification (CTC) translation layer.

Attention OCR [43]: A text recognition method based on the attention mechanism is proposed in it. This method does not require a text box and can be trained end-to-end, which is simpler and more versatile, and greatly surpasses the previous optimal method.

MORAN [44]: The MORAN (Multi-Object Rectified Attention Network) is composed of a multi-object rectification module which is designed for rectifying images that contain irregular text and an attention-based sequence recognition module. Due to the lack of correction link annotations in most character recognition datasets, the MORAN is trained in a weak supervision way, which requires only images and text labels.

DAN [45]: DAN (Decoupled Attention Network) is an end-to-end text recognizer, which is composed of three components: 1) a feature encoder that extracts features; 2) a convolutional alignment module that performs the alignment operation; 3) a decoupled text decoder that makes final prediction. Compared with other networks, DAN decouples the alignment operation from using historical decoding results and achieves SOTA on multiple text recognition tasks in 2020.

LPRNet [53]: LPRNet, consisting of the lightweight Convolutional Neural Network, is a real-time LP recognition system that does not combine RNNs. Meanwhile, LPRNet is a robust network to handle complex tasks including Chinese LPs.

5.2 Implementation Details

In the experiments, we evaluate the proposed model with other state-of-the-art methods on both LP detection performance and LP recognition performance. All training data comes from our proposed LSV-LP dataset, the largest publicly available video based LP dataset. We conduct all experiments on a computer with an Intel Core 3.4GHz CPU, 12GB of RAM and four NVIDIA 1080Ti GPU. Except for the hardware configuration, the batch size of our model is 16, and the maximum epoch is 20.

In the training stage, all three subsets of the dataset will be fed into the network training model. For the detection part, we follow the experimental settings of the official papers. In the recognition part, the LP area is cut out for training and recognition verification. Since some LPs in the dataset are fuzzy, '#' is used instead. This type of LP participates in the training and testing of the detection stage, but does not participate in the recognition stage. Here, we do not conduct experiments on joint training of recognition and detection. This means that our recognition result is independent of the detection module, and the prediction result of the detection model is not used.

5.3 Results and Analysis on the Validation Set

Detection Accuracy Metric. In the test stage of the experiment, Intersection-over-Union (IoU) is used as the evaluation criterion for positive and negative samples of the predicted results according to the test standard of object detection. The predicted result is considered to be true positive (TP) sample only if its IoU with the ground-truth bounding box is more than 0.5. The rest of the prediction results are false positive (FP) samples. In order to better compare the results of various object detection methods on LSV-LP, we evaluate Precion, Recall, Average Precion (AP) and F-score. The formulas of various evaluation criteria are shown as follows:

$$Precision = \frac{N_{TP}}{N_{predict}}, \tag{18}$$

$$Recall = \frac{N_{TP}}{N_{gt}}, \tag{19}$$

$$Fscore = \frac{\left(1+\alpha^2\right) PR}{\alpha^2 \left(P+R\right)}, \tag{20}$$

where N_{TP} denotes the number of the true positive samples. $N_{predict}$ denotes the number of the predict results and N_{gt} represents the number of the ground-truth bounding boxes. For F-score, α represents the parameter that can reconcile average precision and recall. In the experiments, α is set to 1, namely F1-score. The runtime indicates the efficiency to process a single image (not include the processing of NMS). Due to the large data set and different image sizes, we first detect each frame to get the total running time after processing, and then calculate the average running time of each frame based on the total number of frames. All running times in the experimental results are in milliseconds.

Detection Results and Analysis. We test 7 algorithms, and the results of the detection algorithm are shown in Table 4. YOLOv3 achieves the best performance of 12ms by virtue of its model's fast running mechanism. At the same time, the F-score criterion combining recall and precision achieves a relatively good effect of 67.48%. Faster-RCNN is more sensitive to small targets (LPs, in this case) than the YOLO algorithm, with an F-score of 69.14%. The runtime is longer and requires 58ms. Theoretically, DFF and FGFA combine the optical flow module to fuse the characteristics of adjacent frames, and they accelerate the model and improve the accuracy of video object detection respectively. However, due to the small proportion of LP in the input image, the direct application of the two algorithms achieves poor results. The official version of FGFA by default combines the features of 20 frames adjacent to the examined frame, which causes a lot of noise to the examined frame, and the runtime also increases dramatically, requiring 178ms to achieve only 48.35% of the F-score. The LPDNET proposed in 2020 achieves good results in F-score, with the runtime of 42ms and F-score of 72.18%. Since the official code does not provide training modules, we utilize the provided pre-training model to conduct tests on the proposed dataset while do not fine-tune them. It is clear from Table 4 that our method, MFLPR, achieves the best results on multiple subsets. However, it does not perform well in the *move vs. static* subset. The possible reason is that the frames before and after the examined frame we use have a large offset in the background of the *move vs. static* scene, and it is difficult to play a supplementary role in the examined frame.

Recognition Accuracy Metric. Unlike AOLP, UFPR-ALPR, and SSIG-Segplate, our dataset contains Chinese provincial characters in addition to English letters and numbers. It has the following 3 different characteristics: 1) Compared with letters and numbers, Chinese fonts are more square, and the space inside the font is smaller, which makes the distinction between Chinese fonts very difficult when they are seen from a far distance; 2) Chinese characters only exist at the first position of the LP, which is more distinctive from the random distribution of numbers and letters; 3) Since these Chinese LP datasets contain Chinese characters, other datasets that do not contain Chinese characters are unable to perform model transfer testing and must be retrained. Therefore, we select two test indicators, Accuracy_6C and Accuracy_7C, which respectively represent the probability of correct recognition of all LPs without considering Chinese characters and the probability of correct recognition of all characters. Because some LPs in the dataset could not show the specific characters, they are replaced by #. In the recognition stage, we ignore such LPs and do not include them in the training and testing. The runtime of the recognition phase is calculated in the same way as the detection phase.

Recognition Results and Analysis. In the recognition stage, we test five recognition algorithms, including CRNN [42], AttentionOCR [43], MORAN [44], DAN [45], and LPRNET [53], with the final results shown in Table 5. As can be seen from the results, DAN achieves the best results in each subset, not including the running time. The performance of the two subsets *static vs. move* and *move vs. move* is relatively similar, mainly because their shooting background is driving on the road. However, the LPs in the *move vs. static* subset are largely deformed and distorted, making it

TABLE 4
The detection results of various indicators on the LSV-LP dataset.

Method	LSV-LP												
	static vs. move			*move vs. static*			*move vs. move*			average			
	precision	recall	F-score	precision	recall	F-score	precision	recall	F-score	precision	recall	F-score	Runtime
EfficientDet [50]	51.98	88.65	65.53	73.14	82.38	77.49	58.86	80.92	68.15	59.89	83.59	69.78	62
YOLOv3 [51]	50.58	86.45	63.82	72.66	95.78	**89.49**	51.71	85.52	64.45	54.79	87.84	67.48	**12**
SSD512 [29]	50.77	77.15	61.24	32.36	97.11	48.54	52.18	63.03	57.09	44.98	80.12	57.62	21
Faster-RCNN [49]	37.81	82.06	51.77	70.15	**98.07**	81.79	55.66	76.20	64.33	57.60	86.45	69.14	58
DFF [47]	30.56	58.96	40.25	46.27	84.98	59.91	73.51	67.95	70.62	48.93	68.69	57.15	46
FGFA [46]	19.70	29.78	23.71	35.17	49.23	41.03	**74.73**	67.69	71.00	44.60	52.79	48.35	178
LPDNet [52]	**53.78**	83.75	65.50	**78.51**	63.32	70.10	67.63	84.56	75.15	**68.99**	75.69	72.18	42
MFLPR-Net	51.91	**94.77**	**67.07**	74.33	89.42	81.18	70.44	**86.31**	**77.57**	67.89	**89.47**	**76.7**	31

TABLE 5
The recognition results of various indicators on the LSV-LP dataset.

Method	LSV-LP								
	static vs. move		*move vs. static*		*move vs. move*		average		
	Accuracy_6c	Accuracy_7c	Accuracy_6c	Accuracy_7c	Accuracy_6c	Accuracy_7c	Accuracy_6c	Accuracy_7c	Runtime
AttentionOCR [43]	37.3	36.3	19.8	18.5	58.3	56	44.39	42.68	9.3
CRNN [42]	72.94	71.37	51.20	45.37	65.14	62.55	64.54	61.57	0.6
MORAN [44]	18.22	17.87	8.91	8.91	48.94	48.64	40.3	39.98	4.8
LPRNet [53]	74.12	71.85	48.79	44.51	62.1	59.38	62.89	60.03	**0.26**
DAN [45]	78.17	76.34	57.47	54.39	73.18	71.62	71.35	69.40	1.7
MFLPR-Net	**80.29**	**78.57**	**71.21**	**69.23**	**75.5**	**74.31**	**75.99**	**74.49**	1.8

difficult for attention-based methods such as AttentionOCR and MORAN to achieve satisfactory results. The possible reason is that the clipped area is a rectangular box, while the LP is actually a quadrilateral. In the case of deformation, the LP has a large tilt, and it is difficult to learn the parameters of attention to perform well. LPRNet is an open-source LP recognition method. Its network structure is very simple without combining RNN, so it has sufficient advantages in running time. It achieves an average of 0.26ms to process recognition, that is, the FPS exceeds 3000, which can fully meet the real-time requirements in the LPDR system. As an algorithm proposed in 2020, DAN combines bidirectional encoders and alignment operations to achieve the best results. Due to the affine transformation process, the overall recognition rate of our proposed method has been improved, and the recognition stage can also meet the real-time requirements.

5.4 Performance Impact Analysis

Compared with other datasets, LSV-LP has a larger scale, more negative samples and more complex backgrounds. According to our experimental results and the characteristics of the dataset, we continue to do analytical experiments that affect the experimental results. We divide the influencing factors into two categories: the impact of data volume on performance, and the influence of the proposed framework. The experimental results verify that it is necessary to propose a large-scale LP dataset in video scenes.

Impact of Data Volume on Performance. Generally speaking, large-scale datasets can improve the robustness of the model to adapt to more complex scenarios. This is exactly the motivation for us to present this dataset. In this section, we deeply study the influence of data volume on model performance. We split the training set into 10 equal parts, and add them to the training set with increasing scale. We still choose precision, recall and F1-score as the criterion for comparison of results. Since the training set does not affect the model test time, we discard the runtime comparison.

The results for detection stage on different volumes of the training data is shown in Fig. 10. We test the results on three subsets and all data, including precision, recall and F-socre. It can be clearly seen that large-scale training data is positive for the model's results. After the data size is 0.6 times larger, the improvement of the data volume to the result is no longer obvious. However, the effect becomes worse at 0.5. The reason is that a large number of noise samples are included in this place. The experiments are also tested in the recognition stage, and the results have not changed much from 72.32% to 74.32% which can be seen as shown in Fig. 11. It is worth noting in the figure that the curve of move2static_7c has a significant decrease, which is mainly because the subset of move vs. static contains a lot of noisy data. The learning curve of move2static_7c and move2static_6c in this subset is quite different, which can also reflect Chinese characters may be quite different from letters and numbers in the recognition training. The volume of data has little influence on recognition, but it has a great effect on detection. In general, a large dataset is necessary.

Influence of The Proposed Framework. In addition to the impact of data on the results, the design of the model is no exception. In this section, we conduct ablation experiments on the model. In the detection stage, we set up a comparative experiment without inter-frame feature propagation and with inter-frame feature propagation. In the recognition stage, we carry out a comparative experiment with no correction module and with correction module.

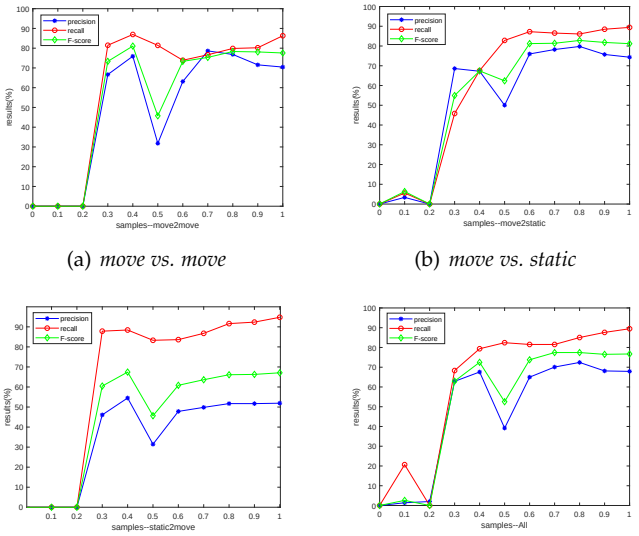
The results of the ablation study are shown in TABLE 6 and TABLE 7. Fusion of adjacent frame information has a positive effect, and its overall F-score is higher than the result without a optical flow module. The same is true for the affine transformation module that the effect of having

TABLE 6
The results of the optical flow module on the detection stage.

Optical Flow Module	*static vs. move* precision/recall/F-score	*move vs. static* precision/recall/F-score	*move vs. move* precision/recall/F-score	average precision/recall/F-score
	53.06/89.73/66.69	**75.76**/**90.03**/**82.27**	63.68/80.18/70.98	66.25/86.33/74.97
✓	51.91/**94.77**/**67.07**	74.33/89.42/81.18	**70.44**/**86.31**/**77.57**	**67.89**/**89.47**/**76.7**

TABLE 7
The results of the affine transformation on the recognition stage.

Affine Transformation	*static vs. move* Accuracy_6c/Accuracy_7c	*move vs. static* Accuracy_6c/Accuracy_7c	*move vs. move* Accuracy_6c/Accuracy_7c	average Accuracy_6c/Accuracy_7c
	76.34/78.17	54.39/57.47	71.62/73.18	69.40/71.35
✓	**78.57**/**80.29**	**69.23**/**71.21**	**74.31**/**75.5**	**74.49**/**75.99**



(a) *move vs. move* (b) *move vs. static*

(c) *static vs. move* (d) average

Fig. 10. The results of the detection stage under different volumes of the training data on the validation dataset. a,b,c are the three subsets of the LSV-LP dataset and d is the average results of the whole dataset.

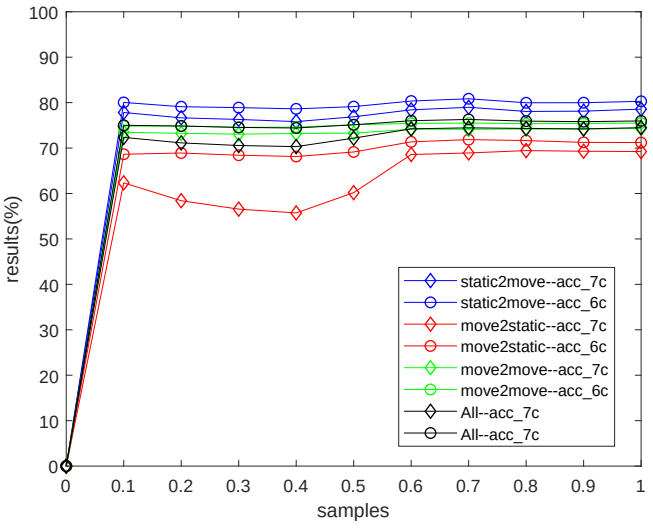


Fig. 11. The results of the recognition stage under different volumes of the training data on the validation dataset.

this module is significantly higher than that of not having it. The experimental results not only confirm the effectiveness of the affine transformation but also verify that the optical flow module brings positive benefits to the overall model.

6 Conclusion and Outlook

In this paper, we propose a large-scale video based LP dataset, namely LSV-LP, which is carefully annotated, including vehicle positioning boxes, points of the LP vertex, and LP numbers. At the same time, according to the shooting scene, it is divided into three subsets *move vs. move*, *move vs. static* and *static vs. move*, which can be applied to the research of LPDR in the video scene. The large data scale (400k frames), the diversity of data (three different categories) and the detailed annotation make LSV-LP a valuable dataset for LPDR systems. Based on LSV-LP, we present a novel framework named MFLPR-Net, which combines the features extracted from adjacent frames of the examined frame. Extensive experiments prove that the algorithm can improve the accuracy.

According to the research on the datasets and LPDR algorithms, we summarize the following points for further research:

1) **How to combine the context information of the nearby frames to improve the accuracy and efficiency?** Videos contain dynamic information of time sequences and correlation between frames. Reasonable use of timing information can well enhance the recognition accuracy of key frames. Our model uses the optical flow network to calculate feature differences. Whether there is a better way to combine contextual features is worth exploring. Besides, compared with single images, videos involve the high degree of information redundancy, such as the feature similarity between two adjacent frames is high. The key to realize real-time performance is to prune the model reasonably and improve the efficiency of the model on videos.
2) **How to coordinate speed and accuracy?** Fusing video inter-frame information can improve the accuracy and reduce the speed, while using video redundancy information is just the opposite. According

to different actual tasks, reasonable coordination of speed and accuracy plays an important role in video based LP detection and recognition. Video frames can be skipped to improve the recognition speed, or the information of multiple video frames can be fused to improve the recognition accuracy. On this basis, a good model compression scheme can fully exploit the potential of the dataset.

3) **How to balance detection and recognition when optimizing the system?** An LPDR system is composed of LP detection and LP recognition, and the performance of recognition depends on the performance of detection. When the two parts cannot be guaranteed at the same time, the performance of the whole system will be better if the two stages of detection and recognition are properly coordinated. The detection and recognition stages of the entire LPR system can best be jointly trained, and operations such as RoI (Region-of-Interest) Align can be used to connect the two stages.

In the future work, we will continue to study the above mentioned issues and improve the performance of the LPDR systems in the real world.

REFERENCES

[1] Sergio Montazzolli Silva and Claudio Rosito Jung. License plate detection and recognition in unconstrained scenarios. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 580–596, 2018.

[2] Zhenbo Xu, Wei Yang, Ajin Meng, Nanxue Lu, Huan Huang, Changchun Ying, and Liusheng Huang. Towards end-to-end license plate detection and recognition: A large dataset and baseline. In *Proceedings of the European conference on computer vision (ECCV)*, pages 255–271, 2018.

[3] Lele Xie, Tasweer Ahmad, Lianwen Jin, Yuliang Liu, and Sheng Zhang. A new cnn-based method for multi-directional car license plate detection. *IEEE Transactions on Intelligent Transportation Systems*, 19(2):507–517, 2018.

[4] Syed Zain Masood, Guang Shu, Afshin Dehghan, and Enrique G Ortiz. License plate detection and recognition using deeply learned convolutional neural networks. *arXiv preprint arXiv:1703.07330*, 2017.

[5] Hao Wang, Pu Lu, Hui Zhang, Mingkun Yang, Xiang Bai, Yongchao Xu, Mengchao He, Yongpan Wang, and Wenyu Liu. All you need is boundary: Toward arbitrary-shaped text spotting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12160–12167, 2020.

[6] Cong Zhang, Qi Wang, and Xuelong Li. V-lpdr: Towards a unified framework for license plate detection, tracking, and recognition in real-world traffic videos. *Neurocomputing*, 2021.

[7] Cong Zhang, Qi Wang, and Xuelong Li. Iq-stan: Image quality guided spatio-temporal attention network for license plate recognition. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2268–2272. IEEE, 2020.

[8] Qi Wang, Junyu Gao, Wei Lin, and Xuelong Li. Nwpu-crowd: A large-scale benchmark for crowd counting and localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[9] Rayson Laroca, Evair Severo, Luiz A Zanlorensi, Luiz S Oliveira, Gabriel Resende Gonçalves, William Robson Schwartz, and David Menotti. A robust real-time automatic license plate recognition based on the yolo detector. In *2018 international joint conference on neural networks (ijcnn)*, pages 1–10. IEEE, 2018.

[10] Gabriel Resende Gonçalves, Matheus Alves Diniz, Rayson Laroca, David Menotti, and William Robson Schwartz. Real-time automatic license plate recognition through deep multi-task networks. In *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 110–117. IEEE, 2018.

[11] Rung-Ching Chen et al. Automatic license plate recognition via sliding-window darknet-yolo deep learning. *Image and Vision Computing*, 87:47–56, 2019.

[12] Gee-Sern Hsu, Jiun-Chang Chen, and Yu-Zu Chung. Application-oriented license plate recognition. *IEEE transactions on vehicular technology*, 62(2):552–561, 2012.

[13] Gabriel Resende Gonçalves, Sirlene Pio Gomes da Silva, David Menotti, and William Robson Schwartz. Benchmark for license plate character segmentation. *Journal of Electronic Imaging*, 25(5):053034, 2016.

[14] Gilles Velleneuve Trindade Silvano, Ivanovitch Silva, Vinícius Campos Tinoco Ribeiro, Vitor Rodrigues Greati, Aguinaldo Bezerra, Patrícia Takako Endo, and Theo Lynn. Artificial mercosur license plates dataset. *Data in Brief*, 33:106554, 2020.

[15] Lap Yan Chan, Alessandro Zimmer, Joed Lopes da Silva, and Thomas Brandmeier. European union dataset and annotation tool for real time automatic license plate detection and blurring. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6. IEEE, 2020.

[16] Amir E Ghahnavieh, Mahmoud Enayati, and Abolghasem A Raie. Introducing a large dataset of persian license plate characters. *Journal of Electronic Imaging*, 23(2):023015, 2014.

[17] Gee-Sern Hsu, ArulMurugan Ambikapathi, Sheng-Luen Chung, and Cheng-Po Su. Robust license plate detection in the wild. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

[18] Englishlp database. http://www.vision.caltech.edu/Image_Datasets/cars_markus/cars_markus.tar/.

[19] Linjiang Zhang, Peng Wang, Hui Li, Zhen Li, Chunhua Shen, and Yanning Zhang. A robust attentional framework for license plate recognition in the wild. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[20] Jakub Špaňhel, Jakub Sochor, Roman Juránek, Adam Herout, Lukáš Maršík, and Pavel Zemčík. Holistic recognition of low quality license plates by cnn using track annotated data. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE, 2017.

[21] Shouyuan Yu, Baopu Li, Qi Zhang, Changchun Liu, and Max Q-H Meng. A novel license plate location method based on wavelet transform and emd analysis. *Pattern Recognition*, 48(1):114–125, 2015.

[22] Satadal Saha, Subhadip Basu, Mita Nasipuri, and Dipak Kumar Basu. License plate localization from vehicle images: An edge based multi-stage approach. *International Journal of Recent Trends in Engineering*, 1(1):284–288, 2009.

[23] Runmin Wang, Nong Sang, Rui Huang, and Yuehuan Wang. License plate detection using gradient information and cascade detectors. *Optik*, 125(1):186–190, 2014.

[24] Younghyun Lee, Taeyup Song, Bonhwa Ku, Seoungseon Jeon, David K Han, and Hanseok Ko. License plate detection using local structure patterns. In *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 574–579. IEEE, 2010.

[25] Kaushik Deb and Kang-Hyun Jo. Hsi color based vehicle license plate detection. In *2008 International Conference on Control, Automation and Systems*, pages 687–691. IEEE, 2008.

[26] Zhenjie Yao and Weidong Yi. License plate detection based on multistage information fusion. *Information Fusion*, 18:78–85, 2014.

[27] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[28] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[29] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[30] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[31] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014.

[32] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *Proceedings of*

the IEEE conference on computer vision and pattern recognition, pages 2315–2324, 2016.
[33] Zied Selmi, Mohamed Ben Halima, Umapada Pal, and M Adel Alimi. Delp-dar system for license plate detection and recognition. *Pattern Recognition Letters*, 129:213–223, 2020.
[34] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
[35] CL Philip Chen and Bingshu Wang. Random-positioned license plate recognition using hybrid broad learning system and convolutional networks. *IEEE Transactions on Intelligent Transportation Systems*, 2020.
[36] Yi Wang, Zhen-Peng Bian, Yunhao Zhou, and Lap-Pui Chau. Rethinking and designing a high-performing automatic license plate recognition approach. *IEEE Transactions on Intelligent Transportation Systems*, 2021.
[37] Jiafan Zhuang, Saihui Hou, Zilei Wang, and Zheng-Jun Zha. Towards human-level license plate recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 306–321, 2018.
[38] Sergio Montazzolli Silva and Claudio Rosito Jung. Real-time brazilian license plate detection and recognition using deep convolutional neural networks. In *2017 30th SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*, pages 55–62. IEEE, 2017.
[39] Meng Dong, Dongliang He, Chong Luo, Dong Liu, and Wenjun Zeng. A cnn-based approach for automatic license plate recognition in the wild. In *BMVC*, pages 1–12, 2017.
[40] Pavel Svoboda, Michal Hradiš, Lukáš Maršík, and Pavel Zemcík. Cnn for license plate motion deblurring. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3832–3836. IEEE, 2016.
[41] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. *l*_0-regularized intensity and gradient prior for deblurring text images and beyond. *IEEE transactions on pattern analysis and machine intelligence*, 39(2):342–355, 2016.
[42] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2016.
[43] Zbigniew Wojna, Alexander N Gorban, Dar-Shyang Lee, Kevin Murphy, Qian Yu, Yeqing Li, and Julian Ibarz. Attention-based extraction of structured information from street view imagery. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 844–850. IEEE, 2017.
[44] Canjie Luo, Lianwen Jin, and Zenghui Sun. Moran: A multi-object rectified attention network for scene text recognition. *Pattern Recognition*, 90:109–118, 2019.
[45] Tianwei Wang, Yuanzhi Zhu, Lianwen Jin, Canjie Luo, Xiaoxue Chen, Yaqiang Wu, Qianying Wang, and Mingxiang Cai. Decoupled attention network for text recognition. In *AAAI*, pages 12216–12224, 2020.
[46] Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. Flow-guided feature aggregation for video object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 408–417, 2017.
[47] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2349–2358, 2017.
[48] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5551–5560, 2017.
[49] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
[50] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.
[51] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.
[52] Lpd-end-to-end. https://github.com/chensonglu/LPD-end-to-end.
[53] Sergey Zherzdev and Alexey Gruzdev. Lprnet: License plate recognition via deep neural networks. *arXiv preprint arXiv:1806.10447*, 2018.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. And he is also with the Key Laboratory of Intelligent Interaction and Applications (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, P. R. China. His research interests include computer vision and pattern recognition.

Xiaocheng Lu received the B.E. degree in Northwestern Polytechnical University and is currently pursuing the M.E. degree with the School of Computer Science and the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. And he is also with the Key Laboratory of Intelligent Interaction and Applications (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, P. R. China. His current research interest include deep learning and computer vision.

Cong Zhang received the B.E. degree in Northwestern Polytechnical University and is currently pursuing the M.E. degree with the School of Computer Science and the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. And he is also with the Key Laboratory of Intelligent Interaction and Applications (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, P. R. China. His current research interest include machine learning and pattern recognition.

Yuan Yuan (M'05-SM'09) is currently a Full Professor with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. And she is also with the Key Laboratory of Intelligent Interaction and Applications (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, P. R. China. She has authored or co-authored over 150 papers, including about 100 in reputable journals, such as the IEEE transactions and Pattern Recognition, as well as the conference papers in CVPR, BMVC, ICIP, and ICASSP. Her current research interests include visual information processing and image/video content analysis.

Xuelong Li (M'02-SM'07-F'12) is a Full Professor with the School of Artificial Intelligence, OPtics and ElectroNics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. And he is also with the Key Laboratory of Intelligent Interaction and Applications (Northwestern Polytechnical University), Ministry of Industry and Information Technology, Xi'an 710072, P. R. China.