Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.047236

ARTICLE

CVTD: A Robust Car-Mounted Video Text Detector

Di Zhou[1], Jianxun Zhang[1,*], Chao Li[2], Yifan Guo[1] and Bowen Li[1]

[1]Department of Computer Science and Engineering, Chongqing University of Technology, Chongqing, China

[2]College of Information and Engineering, Jingdezhen Ceramic University, Jingdezhen, China

*Corresponding Author: Jianxun Zhang. Email: zjx@cqut.edu.cn

Received: 30 October 2023 Accepted: 11 December 2023 Published: 27 February 2024

ABSTRACT

Text perception is crucial for understanding the semantics of outdoor scenes, making it a key requirement for building intelligent systems for driver assistance or autonomous driving. Text information in car-mounted videos can assist drivers in making decisions. However, Car-mounted video text images pose challenges such as complex backgrounds, small fonts, and the need for real-time detection. We proposed a robust **C**ar-mounted **V**ideo **T**ext **D**etector (**CVTD**). It is a lightweight text detection model based on ResNet18 for feature extraction, capable of detecting text in arbitrary shapes. Our model efficiently extracted global text positions through the Coordinate Attention Threshold Activation (CATA) and enhanced the representation capability through stacking two Feature Pyramid Enhancement Fusion Modules (FPEFM), strengthening feature representation, and integrating text local features and global position information, reinforcing the representation capability of the CVTD model. The enhanced feature maps, when acted upon by Text Activation Maps (TAM), effectively distinguished text foreground from non-text regions. Additionally, we collected and annotated a dataset containing 2200 images of Car-mounted Video Text (CVT) under various road conditions for training and evaluating our model's performance. We further tested our model on four other challenging public natural scene text detection benchmark datasets, demonstrating its strong generalization ability and real-time detection speed. This model holds potential for practical applications in real-world scenarios. The code is publicly available at: https://github.com/DiZ-gogogo/CVTD.

KEYWORDS

Deep learning; text detection; Car-mounted video text detector; intelligent driving assistance; arbitrary shape text detector

1 Introduction

Intelligent driving assistance algorithms are crucial research topics in the field of intelligent driving and academia. In the field of computer vision, object detection has always been one of the most important and popular foundational research topics, as highlighted by previous work [1]. However, text detection poses a unique challenge, especially when it comes to text within Car-mounted videos. In complex road conditions, there is a wealth of textual information in Car-mounted videos that can significantly influence driver behavior and provide valuable clues [2]. Therefore, extracting text information from Car-mounted videos can assist drivers in making informed decisions.

Compared to text detection in natural scenes, the dynamic road conditions in Car-mounted video scenes introduce additional complexity and are influenced by various factors. Detecting text in Car-mounted videos is more challenging than that in natural street scenes. Car-mounted video images face the following difficulties due to objective factors from the Car-mounted cameras. (1) Distortion Phenomenon: The wide-angle capture of Car-mounted cameras leads to distortion at the edges of the image, causing deformation of the text content. (2) Shaking and Blurring: Pixel instability in Car-mounted video images, caused by motion or weather conditions, results in increased blurriness of text content. As shown in Fig. 1, PAN++ [3] exhibits poor performance in detecting complex background text during motion. Our model, on the other hand, can accurately detect text, including smaller text lines. Our objective is to design a high-precision and real-time Car-mounted video text detector to address the challenges of text detection in Car-mounted videos.

Figure 1: The figure above shows the visual results of text detection for the "GT" (ground truth), PAN++ [3], and CVTD models on the Car-mounted Video Text (CVT) dataset. "GT" refers to the visualized results under the ground truth labels

In recent years, deep learning algorithms for text detection in natural scenes have gradually matured, such as the PAN++ [3] model, which is a simple yet efficient text detection model for natural scenes. However, there is still room for improvement in adapting these algorithms to the specific challenges posed by text detection in Car-mounted videos with dynamic road conditions. Fig. 1 provides visual evidence supporting this assertion.

We introduce a robust **C**ar-mounted **V**ideo **T**ext **D**etector (**CVTD**) that utilizes the lightweight ResNet18 [4] model as the backbone for feature extraction. Lightweight backbone feature extraction networks exhibit good representational capacity, possess fewer parameters, and maintain a simple structure. To effectively capture the position and enhance the feature representation of text in images, we propose the Coordinate Attention Threshold Activation (CATA) module. Inspired by PAN++ [3], we introduce the Feature Pyramid Enhancement Module (FPEM) to address the limited feature representation capabilities of lightweight backbone networks. By combining the FPEM with the CATA module, we design the Feature Pyramid Enhancement Fusion Module (FPEFM) to strengthen and fuse the extracted text features, increasing model depth. The FPEFM module is stackable, with a default configuration of two layers of stacking. Drawing inspiration from DBNet [5], the enhanced features are fed into a text detection post-processing stage, and we design a text activation head. The text activation head generates a Text Activation Map (TAM) to effectively separate text foreground

and background information. Finally, pixel aggregation operations produce the text detection results for Car-mounted video analysis.

In summary, our main contributions can be summarized as follows:

• We proposed a Car-mounted video text detector that effectively combines text positional information and global feature information through the CATA module. This module, incorporated with skip connections in the network architecture, enhances the detection precision and speed of the model.

• The TAM we proposed can better separate the foreground and background of the text, solving the challenge of text segmentation in complex scenes.

• We collected a Car-mounted Video Text (CVT) dataset consisting of 2200 images with different texts. Our model was tested on the CVT dataset as well as four publicly available datasets, and it achieved competitive results.

2 Related Work

In recent years, there have been gradual developments in computer vision based text detection algorithms. The classic text detection algorithm, Connectionist Text Proposal Network (CTPN) [6], takes into consideration the characteristic that text often has a different aspect ratio compared to general object detection targets. When detecting text lines, the algorithm achieves this by detecting multiple text segments with a fixed width and then concatenates these detected segments to form a complete and larger text box. However, similar to general object detection, this algorithm represents text boxes as rectangles, resulting in poor performance when detecting curved or multi-oriented text. Subsequently, with the advancement of deep learning computer vision technologies, anchor-based and anchor-free text detection methods have been continuously proposed for applications such as natural scene text detection, license plate detection, and Car-mounted video text detection. Methods based on deep learning heavily rely on mathematical analysis techniques, such as [7,8], which play driving roles in artificial intelligence technology. In the future, these techniques can be applied to more text detection algorithms.

Some methods are based on anchor-based text detectors. The SegLink [9] text detection algorithm, by detecting local text segments multiple times, introduces angle information for text boxes. It concatenates multiple text segments based on certain rules, achieving good performance in detecting multi-oriented straight text and handling text lines of arbitrary lengths. However, the algorithm still struggles to effectively detect arbitrarily curved text and text lines with large spacing. Based on the general object detection algorithm SSD [10], the TextBoxes [11] algorithm targets the intrinsic characteristics of text lines and employs different scale bounding boxes to detect text lines of varying lengths and widths. It modifies the single-scale input to a multi-scale input. This algorithm achieves better results in detecting horizontal text lines but performs less effectively in detecting multi-oriented text lines. On the other hand, the TextBoxes++ [12] algorithm makes further improvements by introducing quadrilateral regression, enhancing its performance in detecting multi-oriented straight text. The RRPN text detection algorithm, based on Faster R-CNN [13], proposes region proposals with rotation angles, demonstrating good detection performance for multi-oriented straight text. However, these anchor-based text detection algorithms, derived from or modified based on general object detection, face challenges in accurately representing the boundaries of text of arbitrary shapes. As a result, they exhibit lower detection precision for text of any curved shape.

Some methods are based on anchor-free text detectors. PSENet [14] addresses the limitations of anchor-based methods by adopting pixel segmentation. The model predicts the minimum text kernel and uses multi-scale text kernel expansion to gradually enclose the text lines. This method is effective in separating text lines of arbitrary curved shapes. However, the use of multi-scale expansion results in relatively complex post-processing, leading to lower text detection speed. Similarly based on pixel segmentation, PAN [15] proposes a cascaded feature enhancement structure to enhance the feature representation capability of lightweight backbone networks. The core idea involves using pixel aggregation algorithms to reconstruct the predicted text lines. The text detection algorithm DBNet [5] incorporates differentiable binarization operations into the segmentation model, enabling the adaptive prediction of text regions. During model inference, the differentiable binarization branch can be removed, simplifying complex post-processing operations and resulting in a faster detection speed. However, this algorithm exhibits lower text detection precision. PAN++ [3] is a pixel segmentation method based on a text kernel that predicts the minimum text kernel and text regions, distinguishing different text instances through specific rules. While adopting the pixel segmentation approach improves the text detection precision of the model, there is still room for improvement in terms of text detection speed. DBNet++ [16] introduces an adaptive scale fusion module, enhancing the model's representational capabilities at the cost of sacrificing some text detection speed, leading to an improvement in text detection precision. ABPNet [17] utilizes a boundary proposal model to generate rough initial boundaries for text boxes and iteratively refines text boundaries during model training, ensuring accurate enclosures of text lines. However, the iterative process results in higher computational complexity and lower text detection speed. PSND [18] is a specialized model for detecting license plate numbers in parking spaces. Inspired by PAN++ [3] and DBNet [5], it introduces cascaded feature enhancement modules and context attention blocks to address complex license plate detection issues, achieving good detection precision. However, this algorithm has a relatively narrow application scope. In comparison, our CVTD method is a lightweight text detection algorithm based on pixel segmentation. It can detect text instances of arbitrary shapes and exhibits superior generalization, detection precision, and model inference speed compared to other natural scene text detection methods.

3 Proposed Method

3.1 Overall Architecture

Our CVTD model is a deep learning approach based on pixel segmentation, designed to process single-frame images from Car-mounted videos, as depicted in Fig. 2. It utilizes the lightweight ResNet18 [4] backbone network for image feature extraction. ResNet18 is a lightweight deep learning architecture with a residual network structure, allowing the model to perform fast inferences and providing certain image feature extraction capabilities for Car-mounted video text detection. The features extracted through the ResNet18 backbone network include the conv2, conv3, conv4, and conv5 layers, resulting in shallow feature maps with different resolutions, representing the original input image at 1/4, 1/8, 1/16, and 1/32 resolutions, respectively. To address the limitations in feature extraction and model representation capacity of the ResNet18 backbone network, we enhance these shallow feature maps. This enhancement is achieved by stacking two layers of Feature Enhancement Fusion Modules (FPEFM). The CATA module is employed to extract text features corresponding to the shallow feature maps at four different resolutions, as detailed in Fig. 3. The extracted text features are then integrated into the FPEFM module, as depicted in Fig. 4, which takes the output of the CATA module and the shallow features before enhancement. Consequently, it continuously enhances the model's representational capacity and integrates both local and global text features. The entire

CATA module incorporates a residual skip-connection structure, effectively preventing degradation in the model's feature representation capability due to excessive network stacking. Through the combined effects of the FPEFM and CATA modules, our CVTD model simultaneously strengthens shallow features and integrates text features during the feature enhancement process. After multiple stackings of FPEFM modules, the enhanced features are upsampled to a unified 1/4 resolution and concatenated. This concatenated feature is then input into the post-processing stage of Car-mounted video text detection. CVTD feeds the concatenated features into the text detection head and text threshold activation head. Inspired by the PAN++ [3] model, the Text Detection Head generates text kernels, text regions, and instance vectors. The Text Activation Head produces a text threshold map, effectively separating text and non-text regions. The text threshold map and the text kernel, both being single-channel feature maps, are element-wise added. Finally, the text detection output is obtained through Pixel Aggregation (PA) [3] operation, merging the outputs of the text detection head and text threshold activation head to achieve the final output of Car-mounted video text detection.

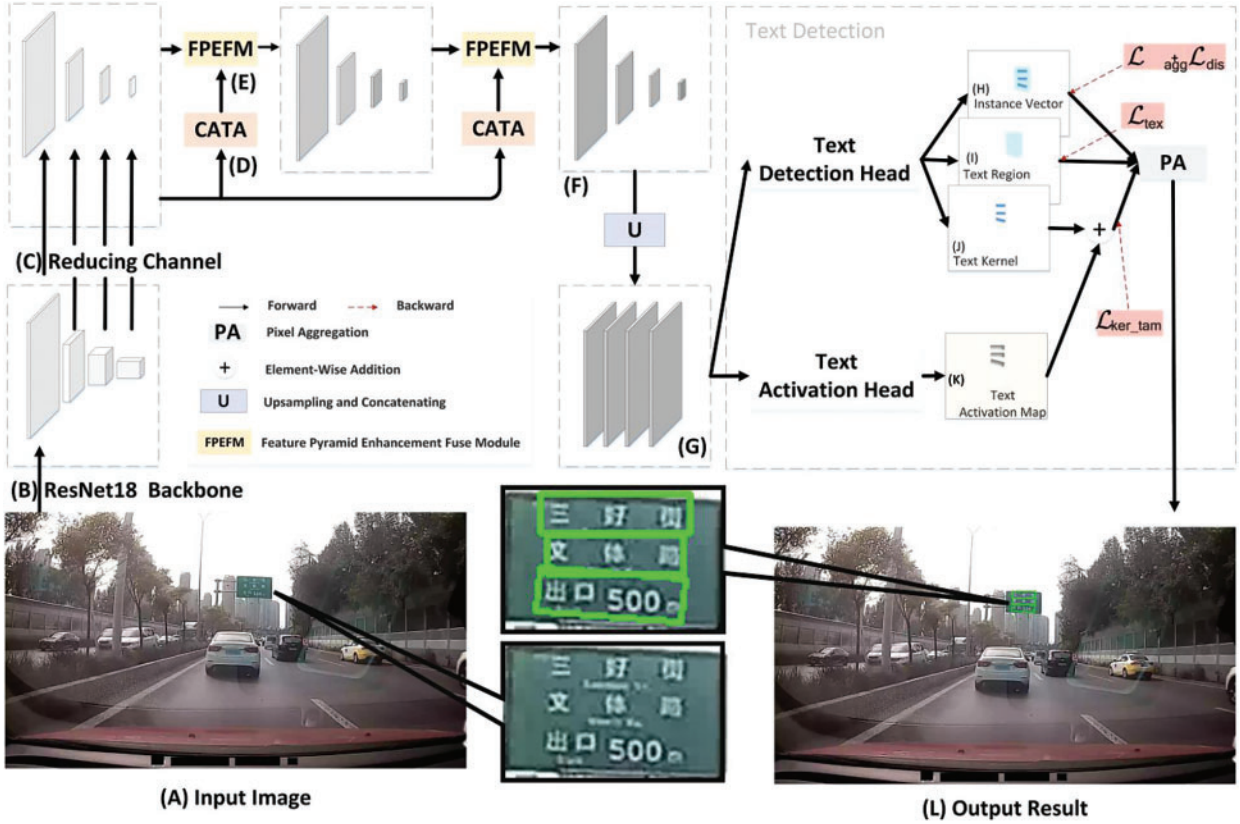

Figure 2: The overall architecture of the CVTD model. ResNet18 is employed as the backbone for feature extraction. The "Reducing Channel" step is utilized to standardize the number of feature channels to 128. Shallow features undergo text feature extraction through the CATA module, as detailed in Section 3.2. The FPEFM module enhances and integrates text features, as described in Section 3.3. Black arrows denote the forward computation of the model, while red dashed lines indicate the gradient back propagation during training

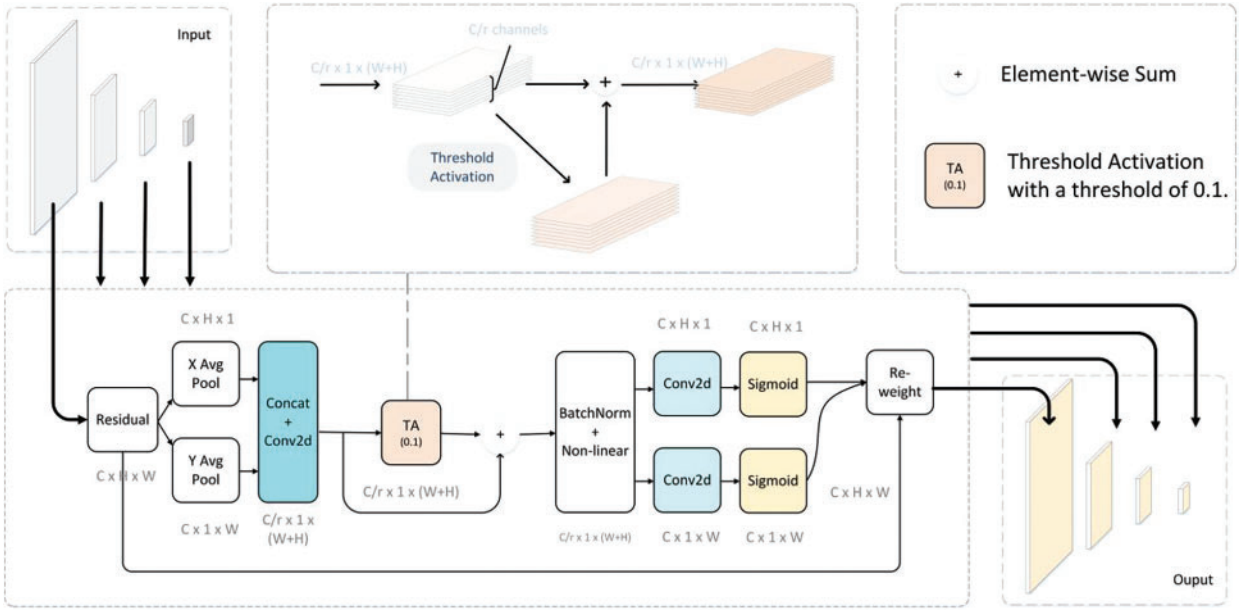

Figure 3: Detail presentation of CATA module. CATA is an adaptive module for text feature extraction. CATA takes inputs of four different resolutions, enabling the backbone shallow feature maps to be more focused on the text regions after passing through the CATA module. "Concat" indicates channel concatenation. "TA" represents the Threshold Activation operation. "Re-weight" signifies the reassignment of feature map weights

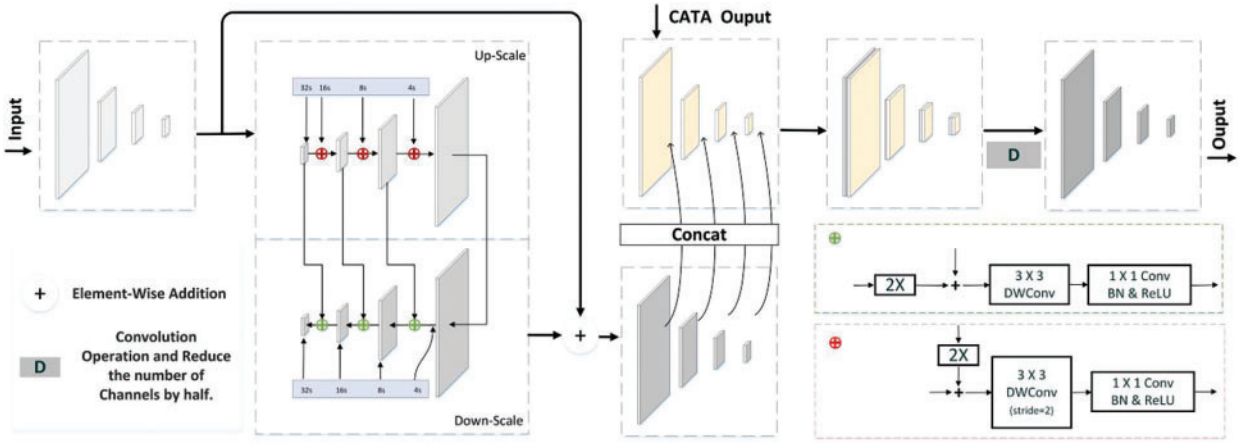

Figure 4: Detailed presentation of the FPEFM model. In this context, "Concat" represents the operation of concatenating the feature maps from the CATA module output and the first-stage output of the FPEFM module along the channel dimension

During the model inference stage, we input a car-mounted video image of size $H \times W \times 3$ (see Fig. 2A) into the lightweight ResNet18 [4] backbone network (see Fig. 2B). The backbone network generates four shallow feature maps with resolutions of 1/4, 1/8, 1/16, and 1/32 of the input image. These feature maps are then processed through 1×1 convolution to uniformly reduce the number of channels to 128 (see Fig. 2C), ensuring that different-scale feature maps contain diverse image features.

In the process of feature enhancement and fusion of text position information, the shallow features undergo two CATA (see Fig. 2D) modules and two stacked FPEFM (see Fig. 2E) modules based on residual skip connections. The output of CATA, which extracts local text features and global position information, becomes part of the input to FPEFM for feature enhancement fusion. The output of the first enhanced features from the FPEFM module becomes the input for the second round of feature enhancement. Finally, the enhanced features are upsampled to the 1/4 size resolution of the input image, resulting in a final feature map with dimensions $H/4 \times W/4 \times 512$ (see Fig. 2G). After text detection post-processing, the final output of Car-mounted video text lines is obtained, assisting in driver decision-making.

3.2 Coordinate Attention Threshold Activation

In the current advanced text detection networks for natural scenes, models employing Transformer Encoder-Decoder structures with position embedding during backbone feature extraction are commonly used. However, the extraction and embedding of position information during feature enhancement are not considered. After the introduction of Coordinate Attention (CA) [19], it has been proven that such operations can make the model pay more attention to position information, which is crucial for text detection in Car-mounted videos.

Therefore, we proposed the Coordinate Attention Threshold Activation (CATA) module for incorporating positional information during feature enhancement in text detection, effectively improving the model's representational capacity. The CATA module receives shallow features extracted from the ResNet18 backbone network, with resolutions of 1/4, 1/8, 1/16, and 1/32. The CATA structure is illustrated in Fig. 3, assuming an input resolution of 1/4 for the feature map. The feature map undergoes horizontal pooling (X-axis) and vertical pooling (Y-axis). The pooled feature maps are concatenated and processed through convolution fusion to extract sensitive and relevant features of interest. For a high-resolution feature map of size 1/4, CATA focuses more on local features, while for a low-resolution feature map of size 1/32, CATA emphasizes the global features of the text. After the extracted features of interest go through the Threshold Activation (TA) operation, sensitive text features are amplified, and the features undergo element-wise addition. Furthermore, the extracted text features are further enhanced in both directions. This is achieved through Conv2d convolution and Sigmoid activation. The Re-weight operation weights the original input feature map, producing high-resolution feature maps for local text features and low-resolution feature maps for global text features. The output of the CATA module serves as the input to the FPEFM module, as depicted in Fig. 4, continuously strengthening the model's representational capacity and enhancing the text detection capabilities in Car-mounted videos.

The calculation process of the CATA module is as follows:

Assuming the given input to the CATA module is a shallow feature map obtained from the feature extraction backbone network, where the resolution of the shallow feature map is $C \times W \times H$. Here, C represents the number of channels in the input feature map, W is the width of the image, and H is the height of the image. Encoding is performed for each channel along the horizontal and vertical directions using a pooling kernel of size $(H, 1)$ or $(1, W)$. In Eq. (1), H denotes the height of x_c, W and denotes the width of x_c. $x_c(h, i)$ represents the encoding of the pixel point x representing the shallow feature along channel C and height h in the feature map. $z_c^h(h)$ is the output of the C th channel with height H can be written as:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i < W} x_c(h, i) \tag{1}$$

$x_c(j, w)$ represents the encoding of the pixel point x representing the shallow feature along channel C and height w in the feature map and $z_c^w(w)$ is the output of the C th channel with width W can be written as:

$$z_c^w(w) = \frac{1}{H} \sum_{0 \le j < H} x_c(j, w) \tag{2}$$

After processing according to Eqs. (1) and (2), the shallow input features obtain corresponding features along the horizontal and vertical directions, forming a pair of text-sensitive feature maps. Subsequently, the feature maps from both directions are concatenated together and input into a convolutional module with a shared 1×1 convolutional kernel, facilitating the fusion of text features in both horizontal and vertical directions. This fusion process reduces the channel dimension to the original C/r. Here, r is a scaling factor. After applying the Sigmoid activation function to the fused feature map, a text-sensitive feature map f with a resolution of $C/r \times 1 \times (W + H)$ is obtained.

$$f = \delta\left(F_1\left(\left[z^h, z^w\right]\right)\right) \tag{3}$$

To further enhance the textual features of the feature map f, the feature map f is input into the Threshold Activation (TA) module. The TA operation T activates the threshold of the feature map f, highlighting the textual features. The activated feature map $T_{0.1}(f)$ is then added to the original feature map f element-wise, resulting in the enhanced feature map t. The default threshold for text activation is set to 0.1.

$$t = T_{0.1}(f) + f \tag{4}$$

Afterward, the feature map t is further enhanced in both directions through a 1×1 convolution, resulting in feature maps F_h and F_w. After applying the σ activation function, attention weights g^h in the height direction and g^w in the width direction are obtained. The Re-weight operation then weights the original input feature map, producing the output of the CATA module.

$$g^h = \sigma\left(F_h\left(t^h\right)\right) \tag{5}$$

$$g^w = \sigma\left(F_w\left(t^w\right)\right) \tag{6}$$

Finally, $x_c(i,j)$ is the input shallow feature map with C channels, and after the above steps, the output is obtained as $y_c(i,j)$. The output $y_c(i,j)$ of the CATA can be written as follows:

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j) \tag{7}$$

The feature map extracted by CATA, containing text-local features and global position information, serves as one of the inputs to the FPEFM feature enhancement and fusion module in the fusion stage.

3.3 Feature Pyramid Enhancement Fusion Module

The Feature Pyramid Enhanced Fusion Module (FPEFM) (see Fig. 2E) and the CATA module together form the basic unit of the Feature Enhancement Network, as shown in Fig. 4.

FPEFM is a Feature Enhancement and Fusion Module that effectively extracts features from Car-mounted video images, enhancing the model's representational capacity. The input to FPEFM consists of two parts: the output of the CATA module, which extracts local text features and global text position information, and the shallow features from the backbone network. The FPEFM module

initially iteratively enhances the pyramid structure of the original image resolution by 1/4, 1/8, 1/16, and 1/32. The first step is the Feature Self-Enhancement stage, where the pyramid's resolution is enhanced from 1/32 to 1/4, and it is element-wise added to the shallow feature map. The Down-Scale pyramid stage enhances the resolution from 1/4 to 1/32 and is element-wise added to the Up-Scale pyramid stage. The second step is the Fusion stage, where the self-enhanced feature maps are fused with the output of CATA, which extracts local text features and global position information. With the help of the residual skip-connection structure in CATA (see Fig. 2D), FPEFM can adaptively optimize the optimal feature enhancement fusion structure, thereby adapting to enhance the model's feature representation capability. The CVTD model predicts the text activation map through the text activation head. The structure of the text activation head is illustrated in Fig. 5. A stronger feature enhancement capability of FPEFM facilitates easier separation of the text activation map into foreground and background components. We experimented with the stacking layers of FPEFM and found that setting it to two layers achieves a balance between Car-mounted video text detection accuracy and inference speed. Therefore, in our experiments, we default to using a two-layer stacked FPEFM structure. Experimental results are presented in Table 1 and Fig. 6.

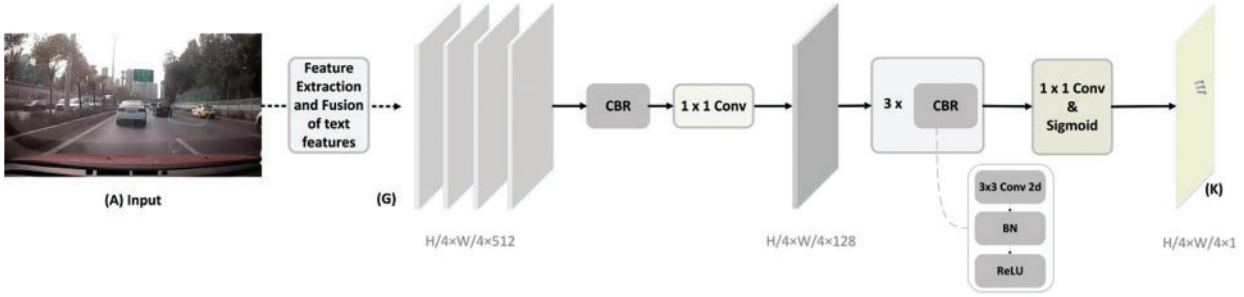

Figure 5: Convolution process of text activation map. "BN" stands for batch normalization. "CBR" is a combination of "3 × 3 Conv 2d", "BN", and "ReLU" operations. The "1 × 1 Conv" is utilized to reduce the number of channels in the enhanced feature map, thereby reducing computational load. The "Sigmoid" activation function is applied to activate the text region

Table 1: Results at CVT and ICDAR2015, stacking different FPEFM layers. The number of experimental stacks is 1–4. "#FPEFM" stands for FPEFM module set the number of stacking 1–4 experimental results. The input size for both datasets is 896 × 896 × 3, respectively

#FPEFM	CVT				ICDAR2015			
	Precision	Recall	F-measure	FPS	Precision	Recall	F-measure	FPS
1	80.3	76.8	78.5	35.2	88.1	77.9	82.7	33.7
2	80.7	77.4	79.0	33.6	88.3	78.6	83.2	30.4
3	80.9	76.4	78.6	29.6	88.3	77.9	82.8	27.8
4	82.6	77.1	79.8	27.3	88.6	77.9	82.9	25.1

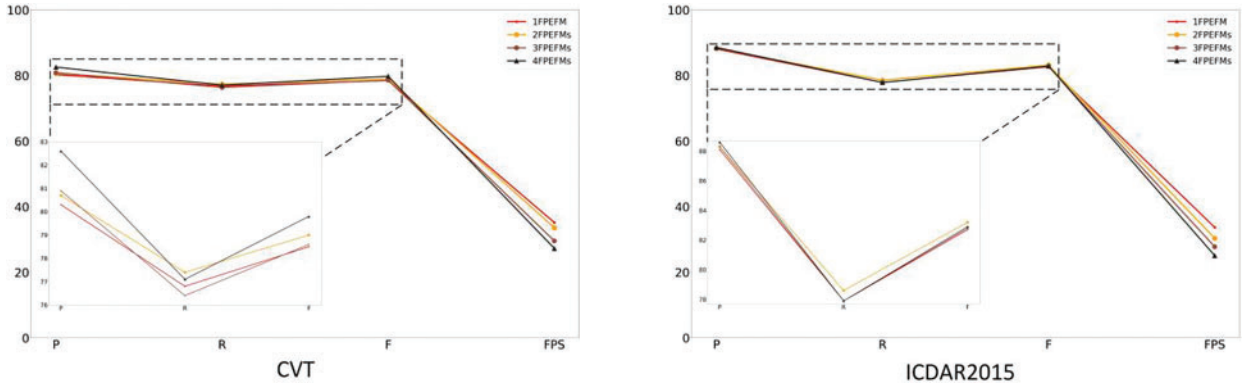

Figure 6: Feature pyramid enhancement fusion module (FPEFM) stacking layers' impact on the visualization results for the CVT and ICDAR2015 datasets

3.4 Text Activation Map

The input image undergoes feature extraction and fusion of text features, resulting in the final enhanced feature map, as illustrated in Fig. 2K. The enhanced feature map has 512 channels. To further fuse features, a series of CBR operations are applied, where CBR comprises a set of operations including 3×3 convolution, Batch Normalization (BN), and ReLU activation. This process is followed by a 1×1 convolution to obtain a feature map with 128 channels, reducing the computational load for model post-processing. The fused feature map is then fed into the model's text activation head. The text activation head undergoes three sets of CBR operations, followed by 1×1 convolution and a Sigmoid activation function. Eventually, a single-channel text activation map is obtained (see Fig. 2K). The Text Activation Map enhances the text regions in Car-mounted video, effectively distinguishing complex text backgrounds. The TAM module exhibits good generalization capabilities for different scenarios, such as rainy weather, nighttime conditions, motion-induced blurriness, and situations where illumination causes minimal contrast differences in the images. The model inference process for the Text Activation Map is illustrated in Fig. 5. Threshold activation is employed during the generation of the Text Activation Map, effectively extracting text regions of interest from the model and demonstrating resistance to background interference. The generated Text Activation Map is highly sensitive to text content, and pixel aggregation is used to improve text line detection performance. Additionally, the information in the Text Activation Map is fused with the text kernels generated by the text detection head (see Fig. 2J) through pixel-wise addition. Through optimization in the loss function, the text detection head and text activation head complement each other, enabling the CVTD model to effectively represent text lines of arbitrary shapes.

3.5 Loss Function

The loss function $\mathcal{L}_{det}$ of our CVTD model can be written as:

$$\mathcal{L}_{det} = \mathcal{L}_{tex} + \alpha \mathcal{L}_{ker\_tam} + \beta \left(\mathcal{L}_{agg} + \mathcal{L}_{dis} \right) \tag{8}$$

where $\mathcal{L}_{\text{tex}}$ is the loss for predicting the text region by the model. $\mathcal{L}_{ker\_tam}$ is the loss computed by the element-wise addition of the text activation map and the text kernel, continuously separating the text region from the non-text region. i represents the pixel point in the input feature map. $\mathcal{L}_{agg}$ loss is used for aggregating the detected text pixels. To balance the importance among $\mathcal{L}_{\text{tex}}$, $\mathcal{L}_{ker\_tam}$, $\mathcal{L}_{agg}$, and $\mathcal{L}_{dis}$, hyper-parameters α and β are set to 0.6 and 0.25, respectively, in all experiments. The text region P_{tex}

and the result of the element-wise addition of the text kernel and the text activation map, denoted as P_{ker_tam}, are supervised using the Dice loss. Thus, the mathematical expressions for $\mathcal{L}_{tex}$ and $\mathcal{L}_{ker\_tam}$ can be expressed as:

$$\mathcal{L}_{ker\_tam} = 1 - \frac{2\sum_i P_{ker\_tam}(i)\, G_{ker\_tam}(i)}{\sum_i P_{ker\_tam}(i)^2 + \sum_i G_{ker\_tam}(i)^2} \tag{9}$$

$$\mathcal{L}_{tex} = 1 - \frac{2\sum_i P_{tex}(i)\, G_{tex}(i)}{\sum_i P_{tex}(i)^2 + \sum_i G_{tex}(i)^2} \tag{10}$$

where $P_{tex}(i)$ and $G_{tex}(i)$ represent the predicted value and ground truth label, respectively, for pixel i in the text region segmented by the model. Similarly, $P_{ker\_tam}(i)$ and $G_{ker\_tam}(i)$ represent the predicted value and ground truth label, respectively, for pixel i in the element-wise addition of the text kernel and the text activation map segmented by the model. To cluster different text lines in Car-mounted videos, it is necessary to increase the gap between classes and reduce the intra-class gap. This is achieved by minimizing the distance between pixels within the same text line and pixels from different text lines. The aggregation loss is used to achieve aggregation for each text line, separating different text lines. The mathematical expression is as follows:

$$\mathcal{L}_{agg} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|T_i|}\sum_{p\in T_i}\mathcal{D}_1(p, KT_i) \tag{11}$$

In Eq. (11), N represents the number of text lines. T_i represents the text region of the i-th text line. KT_i represents the sum of the text kernel and the text activation map of the text line T_i. $\mathcal{D}_1(p, KT_i)$ represents the distance between the text pixel p and the sum of the text kernel and the text activation map KT_i. In the post-processing stage of the model's text detection, the intra-class aggregation of pixel information for Car-mounted video text lines requires separating the instance vectors of text instances from different lines in complex road backgrounds, achieved through the following equation:

$$\mathcal{L}_{dis} = \frac{1}{N^2}\sum_{i=1}^{N}\left(\mathcal{D}_b(KT_i) + \sum_{j=1}^{N}\mathcal{D}_2\left(KT_i, KT_j\right)\right) \tag{12}$$

In Eq. (12), $\mathcal{D}_b(KT_i)$ represents the distance between the sum of the text kernel and the text activation map KT_i and the background. $\mathcal{D}_2\left(KT_i, KT_j\right)$ represents the distance between the sum of the text kernel and the text activation map KT_i and the sum of text kernel and the text activation map KT_j.

4 Experiment

We first introduced our Car-mounted Video Text (CVT) dataset. Furthermore, we evaluated and compared the proposed methods on the CVT dataset and four standard benchmark datasets for natural scene text detection.

CVT: To validate the effectiveness of the proposed method, we selected a Car-mounted video dynamic road condition dataset from the PaddlePaddle platform. We then performed data augmentation and annotation to create a Car-mounted Video Text dataset containing different road conditions. The datasets were annotated using the same format as ICDAR2015. To make the CVT dataset better fit real-world scenes, the data augmentation methods applied to this training data include: (1) random brightness and contrast adjustment, (2) random rotation, and (3) random rotation combined with brightness and contrast adjustment. We randomly split the dataset into 1,540 training images and 660 testing images.

SynthText [20], where the text in the images is synthetic, comprises 800,000 training images. The dataset offers various annotation formats, with each text instance annotated using text strings, word-level bounding boxes, and character-level bounding boxes. There are approximately 8 million synthetic word instances. Consequently, this dataset is utilized as a pre-training dataset for natural scene text detection.

Total-Text [21] is a benchmark dataset of arbitrarily curved shapes, comprising 1,225 training images and 300 test images. It includes different text instances with horizontal, slanted, and curved orientations. This dataset provides detailed annotations, including polygon annotations and word-level annotations.

CTW1500 [22] is also a benchmark dataset of arbitrarily curved shapes, with more text instances. It consists of 1,000 training images and 500 test images. This dataset provides a large number of cropped images of arbitrarily curved text instances and serves as a benchmark for detecting both horizontal and multi-oriented text.

ICDAR2015 [23] is a multi-oriented straight text dataset where the annotated text regions provide detailed word-level annotations. The dataset consists of a total of 1,500 images, comprising 1,000 training images and 500 test images. Sourced from natural street scenes, this dataset is widely used as a benchmark for natural scene text detection.

MSRA-TD500 [24] is a classic dataset for multi-oriented straight text, including both Chinese and English text with line-level annotations. It consists of only 500 natural scene images (300 for training and 200 for testing). Due to its relatively small training set, the **HUST-TR400** [25] dataset, comprising 400 images, is used as additional supplementary data.

4.1 Implementation Details

To evaluate the generalization ability of our model, we conducted experiments on the CVT dataset and four challenging public datasets: CTW1500 [22] and Total-Text [21] datasets, which contain text of arbitrary shapes, and ICDAR15 [23] and MSRA-TD500 [24] datasets, which contain text in multiple orientations. Our CVTD model was pre-trained on the SynthText [20] dataset and then fine-tuned. All models were trained using the SGD optimizer on a single A6000 GPU. The initial learning rate $lr_{initial}$ was set to 1×10^{-3}, and the learning rate adjustment strategy was as follows. Where *iter* represents the current iteration number, and *max_iter* represents the maximum iteration number. Through learning rate iteration reduction, the model fluctuation gradually decreases.

$$lr = lr_{initial} * \left(1 - \frac{iter}{max\_iter}\right)^{0.9} \tag{13}$$

4.2 Experiment of the Stack Number of FPEFMs

By varying the stacking layers of FPEFM from 1 to 4, we conducted experiments to observe and analyze the impact of stacking layers on the final detection performance and speed. As shown in Table 1, in the CVT dataset, Precision increases with the number of stacks, especially with two stacks, yielding the highest Recall. F-measure is highest with four stacks. As the number of stacks increases, FPS decreases gradually. We found that on the publicly available ICDAR2015 dataset, the Precision increases with the increase in stacking layers, while Recall and F-measure achieve the highest scores when the stacking layers are set to 2. However, the frames per second decrease with the increase in stacking layers. To balance the detection precision and speed of the CVTD model, our experiments default to setting the stacking layers to 2.

4.3 Ablation Study

To verify the effectiveness of CATA and TAM, we compared the performance of Car-mounted Video Text (CVT) detection before and after introducing these two modules in the detection network architecture. Ablation studies were performed on CVT, Total-Text [21] and ICDAR2015 [23] datasets to demonstrate the validity of our proposed CATA and TAM modules. Detailed experimental results of CVT dataset are shown in Table 2. Table 3 displays the comparative experimental results of the CVT dataset. Detailed experimental results of Total-Tex and ICDAR2015 datasets are shown in Table 4.

Table 2: Detection results on CVT. The detection results were obtained using different settings of CATA and TAM. "CATA" represents coordinate attention threshold activation, "2FPEFMs" means stacking two layers of FPEFM modules, and "TAM" represents text activation map, respectively

	CVT		
	Precision	Recall	F-measure
Baseline	78.1	76.2	77.1
Baseline + 2FPEFMs + CATA	80.3	76.0	78.1
Baseline + 2FPEFMs + TAM	79.2	77.9	78.6
Baseline + 2FPEFMs + TAM + CATA	80.7	77.4	79.0

Table 3: Text detection results on CVT. "Scale" indicates the scale of the test image, where "L" indicates that the long side is fixed, and "S" indicates that the short side is fixed. "P", "R", and "F" indicate precision, recall, and F-measure, respectively

Method	Scale	Backbone	External	CVT			
				P	R	F	FPS
PAN++ [3]	S: 896	ResNet18	–	80.4	75.8	78.0	22.9
DBNet [5]	736 × 1280	ResNet18_vd	√	71.1	68.8	69.9	12.6
PSENet [14]	S: 736	ResNet50_vd	√	78.2	76.6	77.4	13.2
	S: 896	ResNet50_vd	√	77.9	78.7	78.3	11.7
Fast [26]	S: 896	TextNet-B	√	80.7	**79.1**	79.7	31.2
EAST [27]	S: 512	ResNet18_vd	√	75.9	74.1	75.0	15.9
	S: 896	ResNet50_vd	√	80.0	76.3	78.1	17.0
CVTD-CATA	S: 896	ResNet18	–	80.3	76.0	78.1	35.0
CVTD-TAM	S: 896	ResNet18	–	79.2	77.9	78.6	36.1
CVTD (ours)	S: 736	ResNet18	–	79.9	77.0	78.4	**48.6**
	S: 736	ResNet18	√	79.8	77.6	78.7	**48.6**
	S: 896	ResNet18	–	80.7	77.4	79.0	33.6
	S: 896	ResNet18	√	**81.6**	78.8	**80.1**	33.6

Table 4: Detection results on ICDAR2015 and total-text. The detection results were obtained using different settings of CATA and TAM. "CATA" represents coordinate attention threshold activation, "2FPEFMs" means stacking two layers of FPEFM modules, and "TAM" represents text activation map. "P", "R", and "F" represent precision, recall, and F-measure, respectively

	ICDAR2015			Total-Text		
	P	R	F	P	R	F
Baseline	86.7	78.4	82.3	86.9	78.6	82.6
Baseline + 2FPEFMs + CATA	87.3 (+0.6)	78.8 (+0.4)	82.8 (+0.5)	87.1(+0.2)	79.9 (+1.3)	83.4 (+0.8)
Baseline + 2FPEFMs + TAM	87.1 (+0.4)	**79.4 (+1.0)**	83.1 (+0.8)	86.9	78.9 (+0.3)	82.7 (+0.1)
Baseline + 2FPEFMs + TAM + CATA	**88.3 (+1.6)**	78.6 (+0.2)	**83.2 (+0.9)**	**89.0 (+2.1)**	**80.0 (+1.4)**	**84.3 (+1.7)**

Effectiveness of CATA: To integrate text position information and global modeling information using a lightweight feature extraction network, we use the CATA module to extract text position information and enhance global feature representation. We visualized the extraction of text features by the CATA module on the CVT, Total-Text, and ICDAR2015 dataset, demonstrating the effectiveness of CATA's sensitivity to text features. The visualization results of the CATA module for feature maps of different resolutions are shown in Fig. 7.

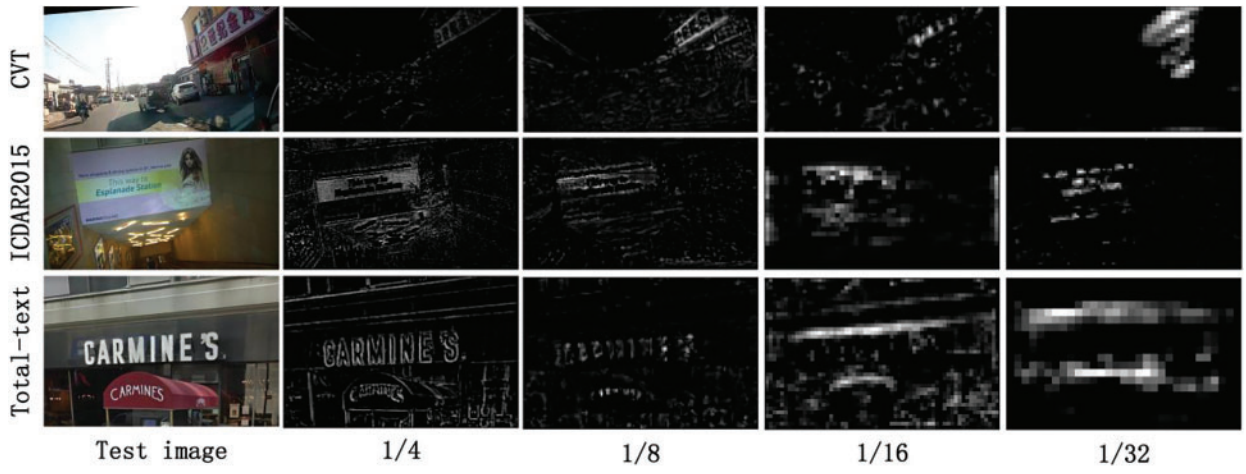

Figure 7: We utilize the class activation map (CAM) technique to visualize the four differently sized features input into the CATA module. It can be observed that with the use of CATA, the CVTD model focuses more on local text features in the 1/4 resolution feature map. The 1/32 resolution feature map pays more attention to the global positional features of the text

The CATA module takes as input shallow feature maps of different resolutions (1/4, 1/8, 1/16, and 1/32) obtained from ResNet18. For the high-resolution feature map with a size of 1/4, CATA focuses more on local features. For low-resolution feature maps with sizes 1/8 and 1/16, CATA shifts its attention from local to global text features. Finally, for the low-resolution feature map with a size

of 1/32, CATA emphasizes global text features. The information extracted by CATA is fused into the FPEFM feature enhancement module. As shown in Table 2, when our model integrates text position information and global information, CATA improves the F-measure by 1%. As shown in Table 4, on the other two publicly available datasets ICDAR2015 [23] and Total-Text [21], CATA improves the F-measure by 0.5% and 0.8%, respectively. As shown in Table 3, compared with other methods, our model achieves the best performance in terms of Precision and F-measure.

Effectiveness of TAM: Due to the special characteristics of text pixels, the ability to effectively determine the category to which a text belongs, the text area, or the non-text area is key to detecting text. The text activation map generated by the text activation head, specifically the TAM module, can effectively enhance the Precision of the model's detection. By using a text activation map to enhance the perception of text regions, we can separate text foreground and background information. As shown in Table 2, our model's designed TAM module improves F-measure by 1.5% on the CVT dataset. As shown in Table 4, on the publicly available datasets ICDAR2015 [23] and Total-Text [21], the F-measure is improved by 0.8% and 0.1%, respectively. On the CVT dataset, when both modules were introduced into our model, the F-measure improved by 1.9%. As shown in Table 3, our method achieves better results.

4.4 Evaluation on CVT

We conducted experiments on the CVT dataset to verify the effectiveness of our proposed method. Our method achieves high Precision, Recall, and F-measure in detecting Car-mounted video text while maintaining real-time detection speed. The detailed experimental results are shown in Table 3. We also provide visual comparisons with several excellent algorithms, as shown in Fig. 8. Our approach achieved the highest precision and F-measure values, as well as real-time detection speed. At a scale size of 896, we achieved a Precision of 81.6%, an F-measure of 80.1%, and an inference speed of 33.6 frames per second (FPS). Under the same experimental settings, compared to PAN++ [3], our method outperformed with a 1.2% higher Precision, 3% higher Recall, and 2.1% higher F-measure. Fast, a network architecture that was carefully designed for text detection, achieved 0.2% higher Recall than our method. However, our method outperformed Fast [26] in terms of detection speed, precision, and F-measure. When setting the scale size to 896, the maximum detection speed reached 48.6 FPS, which is sufficient for real-time detection of text in industrial applications such as Car-mounted video.

Through the analysis of the visual results of the CVTD model and other algorithms on the CVT dataset, as shown in Fig. 8, our model can accurately detect Car-mounted video text in nighttime scenes. There is no "truncation" phenomenon in the detection of text lines compared to the PAN++ [3] model, as shown in the detection results of the PAN++ model in the second row of Fig. 8. For distorted images (shown in the third row of Fig. 8), scenes with strong light (shown in the fourth row of Fig. 8), and rainy scenes (shown in the fifth row of Fig. 8), our model can accurately detect text of various shapes.

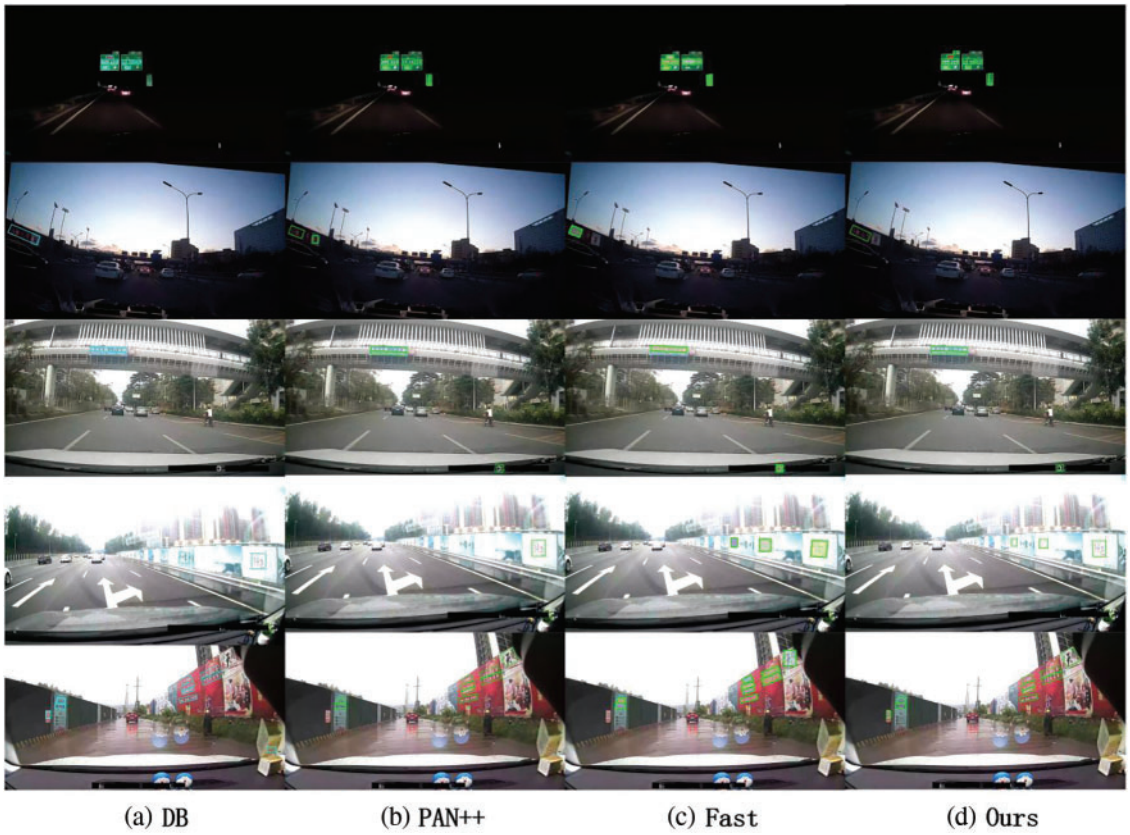

Figure 8: The qualitative comparison results on the challenging CVT dataset include PAN++ [3], DBNet [5], Fast [26], and our CVTD model

4.5 Comparison with Other Methods on the Benchmark Datasets

To validate the generalization ability of our model, we conducted experiments not only on our CVT dataset but also on multiple publicly available benchmark datasets for text detection in natural scenes. We also visualized the detection results for each dataset. The visualization of detection results on the Total-Text [21] and CTW1500 [22] dataset are shown in Fig. 9, and the visualization of detection results on the ICDAR2015 [23] and MSRA-TD500 [24] dataset are shown in Fig. 10. Experimental analysis was conducted on the influence of stacking layers of FPEFM on the ICDAR2015 [23] dataset and the detection numerical results of the stacking layers were visualized, as shown in Fig. 6. Through experiments, when testing on four publicly available dataset, FPEFM had a default stacking layer of 2.

Curved text detection: We conducted experiments on two curved text datasets with our model, as shown in Table 5. Our method achieved an F-measure of 85.5% on the Total-Text [21] dataset, which is a 1.2% improvement compared to the PSND [18] method. On the CTW1500 dataset, our method achieved an F-measure of 84.1%, which is a 0.7% improvement over PSND [18]. Moreover, our method can maintain real-time detection speed. The CVTD model demonstrates high competitiveness on benchmark datasets containing arbitrarily curved text. We visualized the results on these two datasets, as shown in Fig. 9, indicating the model's ability to accurately describe text lines.

Figure 9: The detection visualization results of total-text [21] and CTW1500 [22], both of which are curved text datasets in natural scenes

Figure 10: The detection visualization results of ICDAR15 [23] and MSRA-TD500 [24], both of which are Multi-directional straight text datasets in natural scenes

Table 5: Text detection results on total-text and CTW1500. Methods with "∗" are collected from [3]. "Scale" indicates the scale of the test image, where "L" indicates that the long side is fixed, and "S" indicates that the short side is fixed

Method	Scale	Backbone	External	Total-text				CTW1500			
				P	R	F	FPS	P	R	F	FPS
PAN++ [3]∗	S: 640	ResNet18	√	89.9	81.0	85.3	38.3	87.1	81.1	84.0	36.0
DBNet [5]	S: 800	ResNet18	√	88.3	77.9	82.8	**50.0**	84.8	77.5	81.0	**55.0**
CTPN [6]∗	S: 600	VGG16	–	–	–	–	–	60.4	53.8	56.9	7.1
SegLink [9]∗	768 × 1280	VGG16	–	30.3	23.8	26.7	–	42.3	40.0	40.8	10.7
PSE-1s [14]	S: 1280	ResNet50	√	84.0	78.0	80.9	3.9	84.8	79.7	82.2	3.9
PAN [15]	S: 640	ResNet18	√	89.3	81.0	85.0	39.6	86.4	81.2	83.7	39.8

(Continued)

Table 5 (continued)

Method	Scale	Backbone	External	Total-text				CTW1500			
				P	R	F	FPS	P	R	F	FPS
DBNet++ [16]	S: 800	ResNet18	√	87.4	79.6	83.3	48.0	84.3	81.0	82.6	49.0
PSND [18]	S: 960	ResNet18	√	87.3	81.3	84.3	20.7	84.5	82.3	83.4	20.7
EAST [27]∗	720 × 1280	PVANet	–	50.0	36.2	42.0	–	78.7	49.1	60.4	21.2
ContourNet [28]	720 × 1280	ResNet50	–	86.9	**83.9**	85.4	3.8	83.7	**84.1**	83.9	4.5
TextField [29]∗	768 × 768	VGG16	√	81.2	79.9	80.6	–	83.0	79.8	81.4	–
CRAFT [30]∗	L: 1280	VGG16	√	87.6	79.9	83.6	4.8	86.0	81.1	83.5	7.6
SAE [31]	S: 800	ResNet50	√	–	–	–	–	82.7	77.8	80.1	3.0
SPCNet [32]∗	S: 800	ResNet50	√	83.0	82.8	82.9	4.6	–	–	–	–
FCENet [33]	L: 1280	ResNet50	√	87.4	79.8	83.4	–	85.7	80.7	83.1	–
CVTD	S: 640	ResNet18	–	88.0	80.6	84.1	43.4	83.2	80.7	81.9	35.7
(ours)	S: 640	ResNet18	√	**90.4**	81.2	**85.5**	43.4	**86.8**	81.5	**84.1**	35.7

Multi-oriented text detection: We evaluated our model on the ICDAR2015 [23] and MSRA-TD500 [24] dataset to test its capability for detecting multi-oriented text. As shown in Table 6, it can be seen that our detection results in both datasets have improved the F-measure compared to the DBNet++ [16], with a 1% improvement in ICDAR2015 [23] and a 1.4% improvement in MSRA-TD500 [24]. Compared with other recent algorithms, our method has also achieved competitive results. The detection speed of our method is slightly slower than DBNet [5] but higher than other methods. The CVTD model also demonstrates high competitiveness on benchmark datasets containing multi-oriented straight text. We visualized the results on these two datasets, as shown in Fig. 9. The model maintains high performance in detecting text even in complex scenarios with background interference, such as street scenes.

Table 6: Text detection results on ICDAR15 and MSRA-TD500. Methods with "∗" are collected from [3]. "Scale" represents the scale of the test image, where "L" represents the long side is fixed, "S" represents the short side is fixed, "MS" represents the multi-scale test, and "Max" indicates the maximum size for scale

Method	Scale	Backbone	External	ICDAR2015				MSRA-TD500			
				P	R	F	FPS	P	R	F	FPS
PAN++ [3]∗	S: 736	ResNet18	√	85.9	80.4	83.1	28.2	85.3	84.0	84.7	32.5
DBNet [5]	S: 736	ResNet18	√	86.8	78.4	82.3	**48.0**	**90.4**	76.3	82.8	**62.0**
CTPN [6]∗	S: 600	VGG16	–	74.2	51.6	60.9	7.1	–	–	–	–
SegLink [9]∗	768 × 1280	VGG16	√	73.1	76.8	75.0	–	86.0	70.0	77.0	8.9
DBNet++ [16]	S: 736	ResNet18	√	**90.1**	77.2	83.1	44.0	87.9	82.5	85.1	55.0
ABPNet [17]∗	L: 1024	ResNet50	√	–	–	–	–	86.6	**84.5**	85.6	12.3
EAST [27]∗	720 × 1280	VGG16	–	83.6	73.5	78.2	13.2	87.3	67.4	76.1	–
RRPN [34]∗	L: 1000	VGG16	–	82.0	73.0	77.0	–	82.0	68.0	74.0	–

(Continued)

Table 6 (continued)

Method	Scale	Backbone	External	ICDAR2015				MSRA-TD500			
				P	R	F	FPS	P	R	F	FPS
PixelLink [35]*	768 × 1280	VGG16	–	82.9	81.7	82.3	7.3	81.1	73.0	76.8	3.1
DeepReg [36]*	MS	VGG16	–	82.0	80.0	81.0	–	77.0	70.0	74.0	1.1
SSTD [37]*	704 × 704	VGG16	√	80.2	73.9	76.9	7.7	–	–	–	–
SBD [38]*	Max:1200 × 1600	ResNet50	√	–	–	–	–	89.6	80.5	84.8	3.2
MCN [39]*	512 × 512	ResNet50	√	72.0	80.0	76.0	–	88.0	79.0	83.0	–
RRD [40]*	1024 × 1024	VGG16	√	85.6	79.0	82.2	6.5	87.0	73.0	79.0	10.0
TextSnake [41]*	768 × 1280	ResNet50	√	84.9	80.4	82.6	1.1	83.2	73.9	78.3	1.1
CVTD	S: 896	ResNet18	–	88.3	78.6	83.2	30.4	82.4	82.4	82.4	37.0
(ours)	S: 896	ResNet18	√	87.5	**80.9**	**84.1**	30.4	90.2	83.1	**86.5**	37.0

4.6 Limitation

Our model still has some shortcomings in text detection, especially in images with certain semantic text and too much background noise, as shown in Fig. 11. We will focus on addressing these difficulties in the future.

(a) semantic specific text (b) heavily spaced text (c) truncated text (d) art-like text

Figure 11: Failure samples

5 Conclusions

This paper proposed a new Car-mounted video text detector that uses a lightweight ResNet18 [4] backbone network, making the model very fast in detecting text. Considering the limited extraction capability of the backbone feature extraction network, we proposed the CATA module. The CATA module extracts text features from the input feature maps of four different resolutions. For high-resolution input feature maps, CATA focuses more on local features. For low-resolution feature maps, CATA pays more attention to the global features of the text. Visualizing the output of the CATA module on the CVT, ICDAR2015, and Total-Text datasets demonstrates its effectiveness in extracting text position information and global information. The output of CATA is integrated into the feature enhancement process of the FPEFM module. FPEFM efficiently strengthens features and incorporates text features, continuously enhancing the model's feature representation capability. The enhanced features are then fed into the text detection head and text activation head, utilizing TAM to separate text foreground and background information. In this process, we also constructed a Car-mounted

Video Text (CVT) dataset to evaluate the text detection performance of our proposed Car-mounted Video Text Detector. Experimental results show that the CVTD model achieves the highest Precision, F-measure, and inference speed. The model demonstrates strong text detection capabilities for image distortion, strong light scenarios, and rainy weather scenes. Through generalization performance testing on four public ly available benchmark datasets (Total-Text, CTW1500, ICDAR2015, and MSRA-TD500), the CVTD model exhibits strong competitiveness. Therefore, our proposed CVTD model is a novel, specialized, and efficient Car-mounted video text detection method. CVTD is also a text detector for arbitrary shapes in natural scenes. Its high inference speed enables application in real-world industrial scenarios, contributing to the development of advanced driver-assistance systems.

Acknowledgement: We sincerely appreciate the valuable comments from the editors and every reviewer. Their insights have significantly improved the work presented in this paper, and it is thanks to their professionalism that the quality of the journal has been elevated.

Funding Statement: This work is supported in part by the National Natural Science Foundation of China (Grant Number 61971078), which provided domain expertise and computational power that greatly assisted the activity. This work was financially supported by Chongqing Municipal Education Commission Grants for Major Science and Technology Project (KJZD-M202301901) and the Science and Technology Research Project of Jiangxi Department of Education (GJJ2201049).

Author Contributions: The authors confirm contribution to the paper as follows: Di Zhou completed the model design, experiments, and paper writing. Jianxun Zhang supervised the entire project. Chao Li validated and analyzed the model. Yifan Guo collected data and performed data analysis. Bowen Li debugged the code and analyzed the model's inference speed.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding author, Jianxun Zhang, upon reasonable request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. Akyol, A. Kantarcı, A. E. Çelik and A. Cihan Ak, "Deep learning based, real-time object detection for autonomous driving," in *2020 28th Signal Processing and Communications Applications Conf. (SIU)*, Gaziantep, Turkey, pp. 1–4, 2020.

[2] M. Lyu, J. Song and M. Cai, "A comprehensive method for multilingual video text detection, localization, and extraction," *IEEE Transactions on Circuits and Systems for Video Technology*, Colombo, Sri Lanka, vol. 15, no. 2, pp. 243–255, 2005.

[3] W. Wang, E. Xie, X. Li, X. Liu, D. Liang *et al.*, "Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5349–5367, 2021.

[4] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 770–778, 2015.

[5] M. Liao, Z. Zou, Z. Wan, C. Yao and X. Bai, "Real-time scene text detection with differentiable binarization," in *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 34, no. 7, pp. 11474–11481, 2020.

[6] Z. Tian, W. Huang, T. He, P. He and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Computer Vision–ECCV 2016: 14th European Conf.*, Amsterdam, The Netherlands, Springer, pp. 56–72, 2016.

[7] S. Momani, O. A. Arqub and B. Maayah, "Piecewise optimal fractional reproducing kernel solution and convergence analysis for the Atangana-Baleanu-Caputo model of the lienard's equation," vol. 28, no. 8, pp. 2040007, 2020. https://doi.org/10.1142/S0218348X20400071
[8] S. Momani, B. Maayah and O. A. Arqub, "The reproducing kernel algorithm for numerical solution of van der pol damping model in view of the atangana-baleanu fractional approach," vol. 28, no. 8, pp. 20400010, 2020. https://doi.org/10.1142/S0218348X20400101
[9] B. Shi, X. Bai and S. Belongie, "Detecting oriented text in natural images by linking segments," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 2550–2558, 2017.
[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.*, "SSD: Single shot multibox detector," in *Computer Vision-ECCV 2016: 14th European Conf.*, Amsterdam, The Netherlands, Springer, pp. 21–37, 2016.
[11] M. Liao, B. Shi, X. Bai, X. Wang and W. Liu, "TextBoxes: A fast text detector with a single deep neural network," in *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 31, no. 1, 2017.
[12] M. Liao, B. Shi and X. Bai, "Textboxes++: A single-shot oriented scene text detector," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3676–3690, 2018.
[13] R. Girshick, "Fast R-CNN," in *Proc. of the IEEE Int. Conf. on Computer Vision*, Santiago, Chile, pp. 1440–1448, 2015.
[14] X. Li, W. Wang, W. Hou, R. Liu, T. Lu *et al.*, "Shape robust text detection with progressive scale expansion network," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2019.
[15] W. Wang, E. Xie, X. Song, Y. Zang, W. Wang *et al.*, "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, Seoul, Korea (South), pp. 9336–9345, 2019.
[16] M. Liao, Z. Zou, Z. Wan, C. Yao and X. Bai, "Real-time scene text detection with differentiable binarization and adaptive scale fu-sion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 919–931, 2023.
[17] S. X. Zhang, X. Zhu, C. Yang, H. Wang and X. C. Yin, "Adaptive boundary proposal network for arbitrary shape text detection," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, Montreal, QC, Canada, pp. 1305–1314, 2021.
[18] Y. Zhang, C. Song and M. Xue, "PSND: A robust parking space number detector," in *2022 26th Int. Conf. on Pattern Recognition (ICPR)*, Montreal, QC, Canada, pp. 1742–1748, 2022.
[19] Q. Hou, D. Zhou and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Nashville, TN, USA, pp. 13713–13722, 2021.
[20] A. Gupta, A. Vedaldi and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2315–2324, 2016.
[21] C. K. Ch'ng and C. S. Chan, "Total-Text: A comprehensive dataset for scene text detection and recognition," in *2017 14th IAPR Int. Conf. on Document Analysis and Recognition (ICDAR)*, Kyoto, Japan, IEEE, vol. 1, pp. 935–942, 2017.
[22] Y. Liu, L. Jin, S. Zhang and S. Zhang, "Detecting curve text in the wild: New dataset and new solution," arXiv preprint arXiv:1712.02170, 2017.
[23] D. Karatzas, L. G. Bigorda, A. Nicolaou, S. K. Ghosh, Bagdanov *et al.,* "ICDAR 2015 competition on robust reading," in *2015 13th Int. Conf. on Document Analysis and Recognition (ICDAR)*, Tunis, Tunisia, pp. 1156–1160, 2015.
[24] C. Yao, X. Bai, W. Liu, Y. Ma and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *2012 IEEE Conf. on Computer Vision and Pattern Recognition*, Providence, RI, USA, pp. 1083–1090, 2012.
[25] C. Yao, X. Bai and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4737–4749, 2014.
[26] Z. Chen, J. Wang, W. Wang, G. Chen, E. Xie *et al.,* "FAST: Faster arbitrarily-shaped text detector with minimalist kernel representation," arXiv preprint arXiv:2111.02394, 2021.

[27] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou *et al.,* "East: An efficient and accurate scene text detector," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 5551–5560, 2017.

[28] Y. Wang, H. Xie, Z. J. Zha, M. Xing, Z. Fu *et al.,* "Contournet: Taking a further step toward accurate arbitrary-shaped scene text detection," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pp. 11753–11762, 2020.

[29] Y. Xu, Y. Wang, W. Zhou, Y. Wang and Z. Yang, "TextField: Learning a deep direction field for irregular scene text detection," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5566–5579, 2019.

[30] Y. Baek, B. Lee, D. Han, S. Yun and H. Lee, "Character region awareness for text detection," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, pp. 9365–9374, 2019.

[31] Z. Tian, M. Shu, P. Lyu, R. Li, C. Zhou *et al.,* "Learning shape-aware embedding for scene text detection," in *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 4229–4238, 2019.

[32] E. Xie, Y. Zang, S. Shao, G. Yu, C. Yao *et al.,* "Scene text detection with supervised pyramid context network," in *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 33, no. 1, pp. 9038–9045, 2019.

[33] Y. Zhu, J. Chen, L. Liang, Z. Kuang, L. Jin *et al.,* "Fourier contour embedding for arbitrary-shaped text detection," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Nashville, TN, USA, pp. 3123–3131, 2021.

[34] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang *et al.,* "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, vol. 20, no. 11, pp. 3111–3122, 2018.

[35] D. Deng, H. Liu, X. Li and D. Cai, "PixelLink: Detecting scene text via instance segmentation," in *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 32, no. 1, 2018.

[36] W. He, X. Y. Zhang, F. Yin and C. L. Liu, "Deep direct regression for multi-oriented scene text detection," in *2017 IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, Italy, pp. 745–753, 2017.

[37] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao *et al.,* "Single shot text detector with regional attention," in *Proc. of the IEEE Int. Conf. on Computer Vision*, pp. 3047–3055, 2017.

[38] Y. Liu, S. Zhang, L. Jin, L. Xie, Y. Wu *et al.,* "Omnidirectional scene text detection with sequential-free box discretization," arXiv preprint arXiv:1906.02371, 2019.

[39] Z. Liu, G. Lin, S. Yang, J. Feng, W. Lin *et al.,* "Learning markov clustering networks for scene text detection," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 6936–6944, 2018.

[40] M. Liao, Z. Zhu, B. Shi, G. S. Xia and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pp. 5909–5918, 2018.

[41] S. Long, J. Ruan, W. Zhang, X. He, W. Wu *et al.,* "Textsnake: A flexible representation for detecting text of arbitrary shapes," in *Proc. of the European Conf. on Computer Vision (ECCV)*, pp. 20–36, 2018.