

Article

MBG-PSENet: A PSENet-Based Model for Dense Text Detection in Customs Declarations

Ruiqiang Qiao ¹, Dapeng Yue ¹, Yuwei Gao ¹, Yang Wu ¹ and Tao Bai ^{1,2,*}¹ College of Computer and Information Engineering, Xinjiang Agricultural University, Urumqi 830052, China; 320243443@stu.xjau.edu.cn² Intelligent Agriculture Engineering Research Center of the Ministry of Education, Urumqi 830052, China

* Correspondence: bt@xjau.edu.cn

Abstract

Customs declaration images typically contain dense text, complex table structures, and multi-scale fields, as well as degradation factors such as scan noise, stamp occlusion, and low contrast, making it challenging to locate text regions and separate adjacent text instances. To address the issues of high model complexity, insufficient multi-scale feature fusion, and computational redundancy in the detection heads of the standard PSENet for structured customs declaration text detection, we propose a lightweight dense text detection model, MBG-PSENet. This model retains PSENet's progressive scale expansion mechanism, adopts MobileNetV3 in place of ResNet-50 as the backbone network to reduce the number of parameters and computational overhead, introduces BiFPN to enhance bidirectional multi-scale information exchange between features at different levels, and designs a lightweight detection head, PSEGhostHead, based on GhostModule to reduce redundant convolutional computations. Experiments were conducted on the self-constructed customs declaration text detection dataset CDTD, which contains 879 customs declaration images, including 702 training images and 177 testing images, as well as the ICDAR2015 public dataset. The results show that MBG-PSENet achieves 99.48% Precision, 98.30% Recall, and 98.89% F-measure on the CDTD dataset, with only 1.04 million parameters, 2.05 G FLOPs, a model size of 4.02 MB, and an inference speed of 44.16 FPS. The experimental results demonstrate that MBG-PSENet achieves a good balance among detection accuracy, model size, and computational efficiency, making it suitable for customs declaration OCR applications in resource-constrained scenarios.

Keywords: customs declaration text detection; text recognition; document analysis; MBG-PSENet; PSENet; multi-scale feature fusion; lightweight neural network



Academic Editors: Antonio Fernández-Caballero and Byung-Gyu Kim

Received: 17 June 2026

Revised: 21 July 2026

Accepted: 23 July 2026

Published: 7 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

With the rapid development of international trade and cross-border logistics, customs declarations serve as essential documents for cargo clearance, tax supervision, and trade compliance, and there is an increasing demand for their digital and intelligent processing. In practical customs operations, however, many customs declarations still exist as scanned copies, photographed images, or electronic copies of historical archives. To improve customs clearance efficiency, reduce the cost of manual review, and promote the automation and intelligent processing of document workflows, text detection, text recognition, and information extraction techniques are employed to automatically parse the contents of customs declaration images and convert them into structured information [1,2]. As a

prerequisite for optical character recognition (OCR) [3] and key information extraction [4], the accuracy of text detection directly affects the overall performance of subsequent text recognition, field localization, and semantic parsing. Therefore, developing a text detection method for customs declaration images that achieves high detection accuracy, strong robustness, and good deployment adaptability is of great significance for improving the intelligent processing of structured documents and supporting the automation of customs declaration services.

Compared with natural scene text detection tasks [5], customs declaration text detection involves greater structural complexity and stronger task-specific characteristics. First, customs declarations usually contain dense text regions, regular but complex table structures, and numerous closely arranged text instances, which substantially increases the difficulty of text localization and instance separation. Second, the content of customs declarations often consists of mixed Chinese characters, English letters, digits, and special symbols, with considerable variations in font size, aspect ratio, and character spacing. Third, interference factors such as scanning noise, image blur, uneven illumination, seal occlusion, and handwritten markings further reduce the distinguishability of text regions. In addition, some text in customs declarations is typically small in scale, low in contrast, and distributed within narrow table cells, which can easily lead to missed detections or inaccurate localization. Together, these factors impose higher requirements on existing text detection methods.

In recent years, with the development of deep learning, convolutional neural network-based text detection methods have achieved substantial progress in detection accuracy and generalization ability [6–9]. Among them, PSENet constructs multiple text kernels at different scales and restores complete text regions through a progressive scale expansion strategy, which can effectively alleviate the adhesion problem between adjacent text instances and is therefore representative in dense text detection tasks [10]. However, directly applying PSENet to structured document images such as customs declarations still has certain limitations. Common implementations of PSENet are usually based on relatively heavy feature extraction networks and feature pyramid network (FPN) structures, resulting in a large number of parameters and relatively high computational complexity. Meanwhile, small-scale text, elongated fields, and dense table regions are common in customs declaration images, which places higher demands on the sufficient fusion of shallow spatial details and deep semantic information. When multi-scale feature representation is insufficient, the model is still prone to missed detections or false detections in scenarios involving small-font text localization, adjacent field separation, and complex table-line interference. Therefore, reducing model complexity and enhancing the multi-scale representation capability for structured document layouts while preserving the advantage of PSENet in separating dense text instances is a key issue that needs to be addressed in customs declaration text detection.

To address the aforementioned issues, we propose a lightweight, improved model called MBG-PSENet for dense text detection in customs declarations. Based on the PSENet framework, this model retains the progressive scale expansion mechanism while optimizing the model architecture in three aspects: lightweight feature extraction, multi-scale feature fusion, and a lightweight detection head. These enhancements improve the model's ability to represent complex customs declaration layouts, small-scale text, dense fields, and adjacent text boundaries, while achieving a better balance between detection accuracy and computational efficiency.

The main contributions of this study are summarized as follows:

1. First, to address the lack of publicly available image datasets in current research on customs declaration form text detection, we constructed the CDTD dataset, which is specifically designed for customs declaration form images. Derived from real-world

customs declaration scenarios and processed through image filtering and text region annotation, this dataset captures the characteristics of structured documents, including dense text, complex tables, multi-scale fields, and diverse noise interference, thereby providing data support for the training and evaluation of customs declaration form text detection models.

2. Second, we designed a lightweight customs declaration text detection model, MBG-PSENet, to address the issues of high model complexity, insufficient multi-scale feature fusion, and computational redundancy in the detection head that plague the standard PSENet in structured document scenarios. Through lightweight feature extraction, bidirectional multi-scale feature fusion, and an efficient detection head design, this model maintains its ability to separate dense text instances while enhancing adaptability to small-scale text, narrow fields, and complex table structures, thereby improving the model's deployability in practical customs declaration OCR systems.

3. Finally, we conducted experimental evaluations on our self-constructed CDTD dataset and the ICDAR2015 public dataset, and compared our model with representative text detection models. The experimental results demonstrate that the proposed MBG-PSENet maintains strong detection performance while reducing model complexity and exhibits a certain degree of generalization capability across various scenarios.

2. Related Works

In recent years, general text detection methods based on deep learning have made substantial progress in natural scene text detection and have gradually provided an important technical foundation for text localization tasks in document images, receipt images, and form images [11]. According to the way text regions are modeled, existing methods can be broadly divided into regression-based methods and segmentation-based methods. Regression-based methods usually draw on general object detection frameworks and localize text regions by predicting text bounding boxes, text segments, or geometric parameters. CTPN divides a text line into a series of fixed-width text proposal segments and uses a recurrent neural network to model the sequential context of text, achieving good performance in horizontal or near-horizontal text detection tasks [12]. EAST adopts an end-to-end fully convolutional network to directly predict text region score maps and geometric information, reducing proposal generation and multi-stage processing while improving text detection efficiency [13]. TextBoxes++ designs a single-shot detection architecture suitable for text detection by considering the large variations in text aspect ratios and orientations, thereby enhancing the localization capability for multi-oriented text regions [14]. RRD further improves arbitrary-oriented text detection performance through rotation-sensitive features and a rotation-sensitive regression mechanism [15]. FCENet reformulates arbitrary-shaped text detection as contour regression in the Fourier domain: it predicts a compact set of Fourier signature vectors for each instance and reconstructs the text boundary through an inverse Fourier transformation, enabling smooth and flexible modeling of highly curved text while avoiding complex post-processing [16]. Overall, regression-based methods have the advantages of clear architectures and relatively high inference efficiency. However, their detection performance is often affected by factors such as proposal design, geometric regression accuracy, and non-maximum suppression. In scenarios involving small-scale text, extreme aspect ratios, or closely spaced adjacent text instances, these methods may still suffer from inaccurate boundary localization or missed detections.

Compared with regression-based methods, segmentation-based text detection methods formulate text region localization as a pixel-level classification or instance segmentation problem, allowing them to describe multi-oriented, arbitrarily shaped, and densely distributed text regions more flexibly. PixelLink performs text instance segmentation by

simultaneously predicting text pixels and their neighborhood linkage relationships, thereby effectively improving the separation of adjacent text instances [17]. To further address closely spaced and irregular text, TextField learns a deep direction field in which each text pixel is assigned a vector pointing away from its nearest text boundary; the resulting field provides strong cues for grouping pixels into instances and separating tightly packed text, which is beneficial for densely arranged fields [18]. TextSnake represents each text instance as a series of ordered geometric units distributed along the centerline and predicts attributes such as the centerline, radius, and orientation, which enhances the model's ability to represent curved and irregular text [19]. CRAFT localizes text instances by learning character region probability maps and inter-character affinity relationships, showing strong adaptability in character-level modeling and irregular text detection [20]. PAN adopts a pixel aggregation strategy to improve the efficiency of arbitrary-shaped text detection while maintaining detection accuracy [21]. Building on this pipeline, PAN++ couples a lightweight backbone with a redesigned text-kernel representation and pixel-aggregation post-processing, achieving a more favorable balance between detection accuracy and inference speed for arbitrarily shaped text [22]. DBNet formulates the binarization process as a differentiable module, enabling the joint optimization of text region probability maps and threshold maps, thereby enhancing text boundary learning and simplifying the post-processing procedure [23]. DBNet++ further extends this framework with an Adaptive Scale Fusion module that adaptively aggregates multi-scale features across the feature pyramid, strengthening the representation of text instances with large scale variations while preserving real-time efficiency [24]. In addition, CentripetalText represents each instance through centripetal shifts that pull boundary pixels toward their corresponding text kernels, yielding a compact and robust instance representation that is particularly effective for separating densely distributed text [25]. As a typical segmentation-based text detection method, PSENet predicts multiple text kernels at different scales and progressively restores complete text regions using a progressive scale expansion strategy, giving it clear advantages in separating dense text instances [10].

However, most of the above methods were designed for general text detection tasks and still have certain limitations when directly applied to structured document images such as customs declarations. Customs declaration images usually contain a large amount of small-font text, closely arranged fields, regular table lines, and text regions with markedly different aspect ratios, which place higher requirements on the model's fine-grained boundary perception and multi-scale feature representation capabilities. Although PSENet can effectively separate adjacent text instances, its commonly used framework is typically built on a ResNet-FPN architecture and still suffers from a large number of parameters, high computational complexity, and insufficient fusion between shallow spatial details and deep semantic information. To address these limitations, we propose MBG-PSENet based on the progressive scale expansion mechanism of PSENet, aiming to enhance the model's feature representation capability for complex customs declaration layouts, small-scale text, and dense field boundaries.

3. Materials and Methods

3.1. Dataset

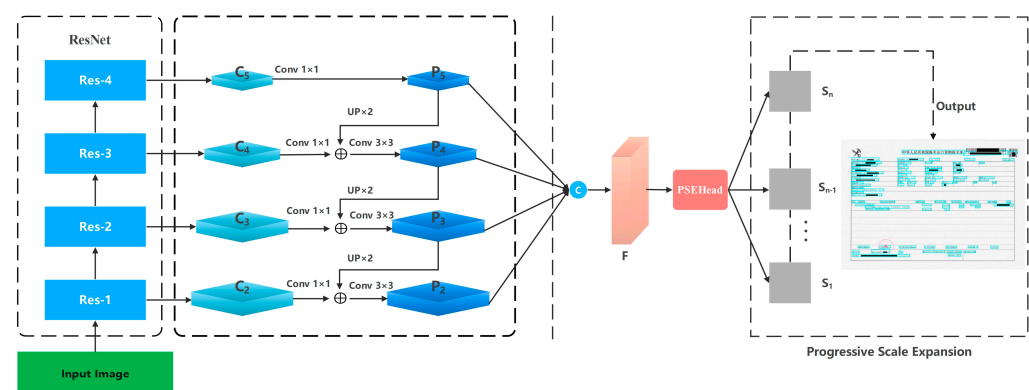
To evaluate the text detection performance of the proposed method in complex scenarios, experiments were conducted on the public ICDAR2015 English scene text dataset [26] and a self-constructed customs declaration dataset. ICDAR2015 is a widely used public benchmark dataset in the field of text detection. It contains 1500 natural scene images, including 1000 images for training and 500 images for testing. The images in this dataset were captured using Google Glass devices and have relatively low resolution. Owing to

the limitations of the imaging device and environmental conditions, the images contain varying degrees of motion blur, illumination variation, and background interference.

To further evaluate the applicability of the proposed method in practical business scenarios, we constructed an in-house customs declaration text detection (CDTD) dataset. The images were selected from real customs declaration processing scenarios, with emphasis on samples whose form structures and text regions were sufficiently visible for reliable annotation. The CDTD dataset contains 879 images and was randomly split at the image level into training and test sets at a ratio of 4:1. The training set comprised 702 images with 71,222 annotated text instances, whereas the test set comprised 177 images with 18,277 annotated text instances. The modal image resolution was 1124×795 pixels. The images were captured using mobile devices under diverse conditions, including different shooting angles, capture distances, illumination conditions, and degrees of paper deformation. Compared with general scene images, customs declaration forms contain more densely arranged text, while table lines and seals may interfere with text localization. Text regions were initially labeled using PPOCRLabel, a semi-automatic graphical annotation tool. The preliminary annotations were then manually reviewed, during which boundary inaccuracies were corrected, previously missed text regions were annotated, and erroneous annotations were removed. This two-stage annotation procedure was used to improve the accuracy and reliability of the dataset and subsequent detection evaluation.

3.2. PSENet Architecture

PSENet (Progressive Scale Expansion Network) is an instance-segmentation-based text detection method. As shown in Figure 1, its overall architecture mainly consists of a backbone feature extraction network, a feature pyramid fusion module, a text-region segmentation branch, and a progressive scale expansion module. The input image is first processed by the backbone network to extract multi-level features. Shallow features preserve rich edge and texture information, whereas deep features provide stronger semantic representations. Subsequently, PSENet adopts a feature pyramid network (FPN) [27] to upsample and fuse features at different scales, thereby generating feature representations that integrate both spatial details and semantic information.



text kernels are then progressively expanded to larger-scale regions in ascending order of scale, ultimately producing complete text detection results. This progressive expansion strategy effectively alleviates the adhesion of adjacent text instances in dense text scenarios, making PSENet suitable for customs declaration forms characterized by closely spaced fields, dense text, and complex tabular structures.

3.3. Overall Network Architecture of MBG-PSENet

To address the high model complexity, insufficient multi-scale feature fusion, and computational redundancy in the detection head of the standard PSENet for customs declaration text detection, we propose a lightweight improved model, namely MBG-PSENet. The overall architecture of the proposed model is shown in Figure 2. MBG-PSENet retains the multi-scale text kernel prediction and progressive scale expansion mechanism of PSENet while introducing architectural optimizations from three aspects. First, MobileNetV3 is adopted to replace the original backbone network, thereby reducing the number of parameters and computational complexity. Second, a bidirectional feature pyramid network (BiFPN) is introduced to perform bidirectional multi-scale feature fusion, which enhances information interaction among features at different levels. Finally, a lightweight detection head based on GhostModule is designed to reduce redundant computation and generate prediction maps for both complete text regions and multi-scale text kernels. The model then applies the progressive scale expansion algorithm to separate text instances and recover text boundaries, thereby achieving the efficient detection of dense text regions in customs declaration forms.

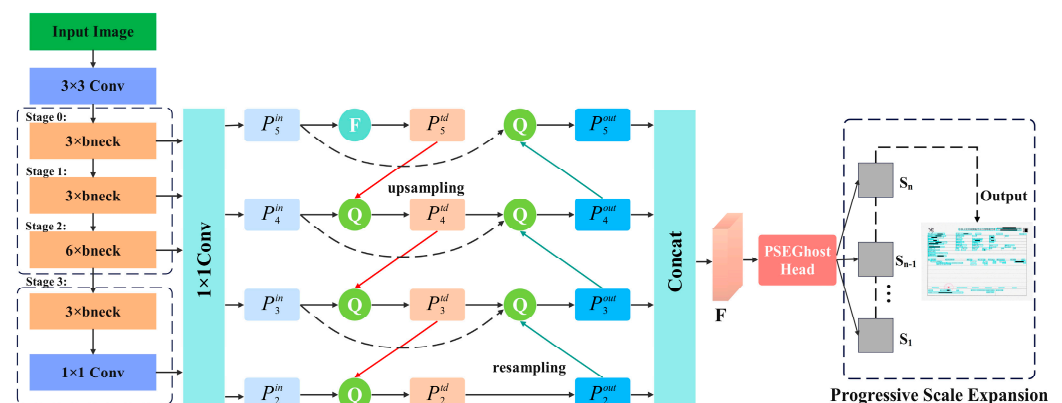


Figure 2. Overall Network Structure of MBG-PSENet.

3.3.1. Improvement of the Lightweight PSENet Backbone Network Based on MobileNetV3

The original PSENet adopts ResNet-50 [28] as its backbone network. Although ResNet-50 provides strong feature extraction capability, it contains a large number of parameters and requires substantial computation, which may result in high inference overhead when processing high-resolution customs declaration images with dense text. Text in customs declaration forms is typically characterized by small font sizes, compact layouts, and a large number of text instances. Therefore, the model needs to preserve its sensitivity to fine-grained strokes and text boundaries while maintaining high computational efficiency. To this end, MobileNetV3 [29] was adopted in this study to replace ResNet-50 as the backbone of PSENet. To meet the lightweight deployment objective of this study, a width multiplier of 0.5 was applied to MobileNetV3-Large, retaining its depth and multi-scale feature-extraction hierarchy while approximately halving the channel widths to balance representational capacity and computational efficiency. The network configuration of MobileNetV3 is presented in Table 1. By incorporating depthwise separable convolutions, inverted residual structures, linear bottlenecks, and the squeeze-and-excitation (SE) atten-

tion mechanism [30], MobileNetV3 reduces redundant computation and model parameters while enhancing the representation of small-scale text regions and local structural features. Therefore, it is more suitable for dense text detection in customs declaration forms. Figure 3 shows the structure of the MobileNetV3 backbone.

Table 1. Network configuration of MobileNetV3-Large (scale = 0.5).

Stage	Input	Operator	Expansion Channels	Output Channels	SE	Nonlinearity	Stride
Conv	$640^2 \times 3$	3×3 Conv	—	8	×	HS	2
Stage 0	$320^2 \times 8$	bneck, 3×3	8	8	×	RE	1
Stage 0	$320^2 \times 8$	bneck, 3×3	32	16	×	RE	2
Stage 0	$160^2 \times 16$	bneck, 3×3	40	16	×	RE	1
Stage 1	$160^2 \times 16$	bneck, 5×5	40	24	✓	RE	2
Stage 1	$80^2 \times 24$	bneck, 5×5	64	24	✓	RE	1
Stage 1	$80^2 \times 24$	bneck, 5×5	64	24	✓	RE	1
Stage 2	$80^2 \times 24$	bneck, 3×3	120	40	×	HS	2
Stage 2	$40^2 \times 40$	bneck, 3×3	104	40	×	HS	1
Stage 2	$40^2 \times 40$	bneck, 3×3	96	40	×	HS	1
Stage 2	$40^2 \times 40$	bneck, 3×3	96	40	×	HS	1
Stage 2	$40^2 \times 40$	bneck, 3×3	240	56	✓	HS	1
Stage 2	$40^2 \times 56$	bneck, 3×3	336	56	✓	HS	1
Stage 3	$40^2 \times 56$	bneck, 5×5	336	80	✓	HS	2
Stage 3	$20^2 \times 80$	bneck, 5×5	480	80	✓	HS	1
Stage 3	$20^2 \times 80$	bneck, 5×5	480	80	✓	HS	1
Stage 3	$20^2 \times 80$	1×1 Conv	—	480	×	HS	1

Note: RE denotes the ReLU activation function, whereas HS denotes the Hard-Swish activation function. In the SE column, ✓ indicates that the squeeze-and-excitation (SE) module is used, whereas × indicates that it is not used. The backbone network ultimately produces feature maps at four scales, with dimensions of $160^2 \times 16$, $80^2 \times 24$, $40^2 \times 56$, and $20^2 \times 80$, which are subsequently fed into the BiFPN for multi-scale feature fusion.

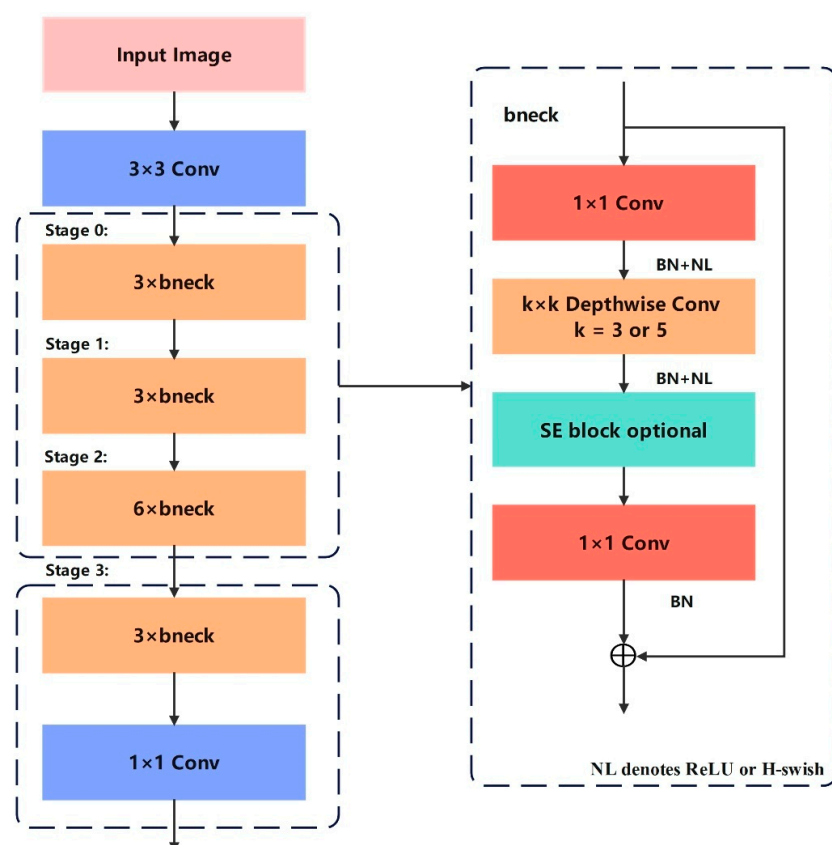


Figure 3. Structure Diagram of MobileNetV3.

MobileNetV3 first applies a standard convolution to the input image for initial down-sampling and low-level feature extraction, after which multiple inverted residual blocks progressively construct multi-scale semantic features. Each inverted residual block typically consists of a 1×1 channel expansion convolution, a $k \times k$ depthwise convolution, an SE channel attention module, and a 1×1 linear projection convolution. For input feature X_l , the l -th inverted residual block can be expressed as:

$$Y_l = \mathcal{P}_{1 \times 1}^{(l)} \left(\mathcal{S}^{(l)} \left(\mathcal{D}_{k \times k}^{(l)} \left(\phi \left(\mathcal{E}_{1 \times 1}^{(l)} (X_l) \right) \right) \right) \right) \quad (1)$$

where $\mathcal{E}_{1 \times 1}^{(l)}(\cdot)$ is the channel expansion convolution, $\mathcal{D}_{k \times k}^{(l)}(\cdot)$ is the depthwise convolution, $\mathcal{S}^{(l)}(\cdot)$ is the SE attention operation, $\mathcal{P}_{1 \times 1}^{(l)}(\cdot)$ is the linear projection convolution, and $\phi(\cdot)$ denotes the nonlinear activation function. When the stride is 1 and the number of input and output channels is equal, a residual connection is incorporated to facilitate feature reuse and stable gradient propagation:

$$X_{l+1} = X_l + Y_l \quad (2)$$

Depthwise separable convolution is the key to reducing the computational complexity of MobileNetV3. Given a convolutional kernel of size $D_k \times D_k$, M input channels, N output channels, and a feature map of size $D_f \times D_f$, the ratio of the computational cost of depthwise separable convolution to that of standard convolution is:

$$\frac{\text{Cost}_{\text{DSConv}}}{\text{Cost}_{\text{Conv}}} = \frac{D_k^2 M D_f^2 + M N D_f^2}{D_k^2 M N D_f^2} = \frac{1}{N} + \frac{1}{D_k^2} \quad (3)$$

MobileNetV3 thus significantly reduces the convolutional computational cost while preserving the ability to model cross-channel interactions and spatial features. To further enhance effective feature selection, MobileNetV3 incorporates the SE mechanism into certain blocks, as illustrated in Figure 4, which generates channel-wise weights through global context modeling to adaptively recalibrate text-relevant features:

$$\tilde{U}_c = U_c \cdot \sigma_{\text{hs}} \left(W_2 \delta \left(W_1 \frac{1}{HW} \sum_{i=1}^H \sum_{j=1}^W U_c(i, j) \right) \right) \quad (4)$$

where $U \in R^{C \times H \times W}$ denotes the input feature map of the SE module, and C , H , and W denote the number of channels, the height, and the width of the feature map, respectively. $U_c \in R^{H \times W}$ represents the c -th channel of the input feature U , and $U_c(i, j)$ denotes the response value of this channel at the spatial position (i, j) . $\tilde{U}_c$ denotes the c -th output feature after channel recalibration. W_1 and W_2 represent the weight parameters of the two 1×1 convolutional layers in the SE module, respectively. Specifically, W_1 is used for channel reduction to decrease the parameter overhead, whereas W_2 is used for channel expansion to restore the original channel dimension. $\delta(\cdot)$ denotes the ReLU activation function, and $\sigma_{\text{hs}}(\cdot)$ denotes the hard-sigmoid activation function, which is used to generate normalized channel attention weights. The generated weights are broadcast to the spatial dimensions of the corresponding channels and multiplied channel-wise with the original input features, thereby adaptively enhancing or suppressing the responses of different channels. Through this mechanism, the network can exploit global contextual information to emphasize key responses related to text strokes, edges, and local structures while suppressing interference from background textures and irrelevant regions. Consequently, the representational capability of MobileNetV3 for dense text features in customs declaration forms is improved.

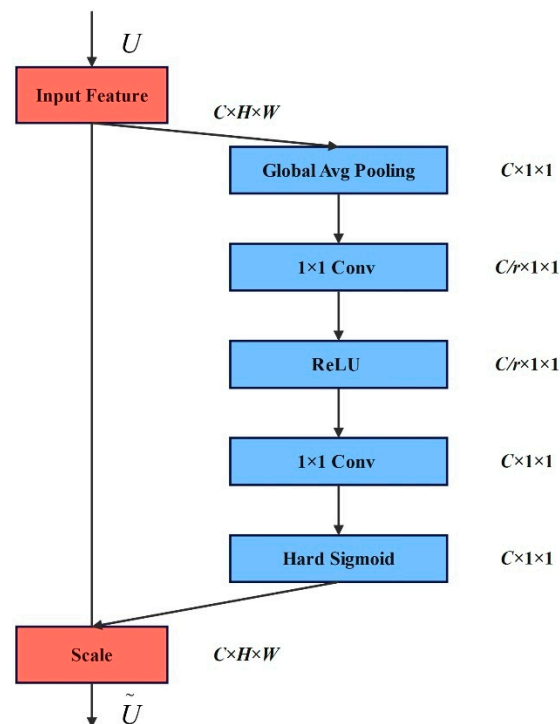


Figure 4. SE Attention Module.

3.3.2. Optimization of the Multi-Scale Feature Fusion Network Based on BiFPN

The original PSENet adopts a standard feature pyramid network (FPN) for multi-scale feature fusion. Its information propagation path follows a strictly top-down unidirectional structure, in which high-level semantic features are progressively upsampled and then simply fused with low-level features by element-wise addition. This design has inherent limitations when handling dense text scenarios in customs declaration forms. On the one hand, the fine-grained spatial location information contained in low-level features cannot be fed back to higher layers, resulting in insufficient semantic alignment among cross-scale features. On the other hand, FPN applies an equal-weight fusion strategy to all input feature layers, failing to distinguish the different contributions of features at different scales to the detection task.

Customs declaration forms commonly contain complex layout characteristics, such as densely arranged characters, extremely small line spacing, and the coexistence of characters with different sizes. Owing to the above limitations, the model has insufficient capability to distinguish the boundaries between adjacent text instances, which may lead to false detections and missed detections. Therefore, this study introduces the bidirectional feature pyramid network (BiFPN) [31] to replace the original FPN module, as shown in Figure 5. By constructing bidirectional cross-scale feature interaction paths and a learnable adaptive weighted fusion mechanism, BiFPN effectively improves the representation quality of multi-scale features.

The multi-scale features output by the backbone network are first passed through 1×1 convolutions to unify their channel dimensions, yielding P_i^{in} . BiFPN enables cross-scale information interaction through two pathways, namely a top-down pathway and a bottom-up pathway. The top-down pathway generates the intermediate feature P_i^{td} , whereas the bottom-up pathway fuses P_i^{in} , P_i^{td} , and the resampled output feature from the lower level to obtain the final output feature P_i^{out} . In Figure 5, F denotes the feature refinement module, which applies depthwise separable convolution, batch normalization, and Swish activation to a single-path input feature. Q denotes the fusion refinement operator, which performs learnable normalized weighted fusion on multi-path input features and then re-encodes

the fused feature through the feature refinement module. Finally, the output features at different scales are resized to a unified resolution and concatenated along the channel dimension to form a comprehensive feature map.

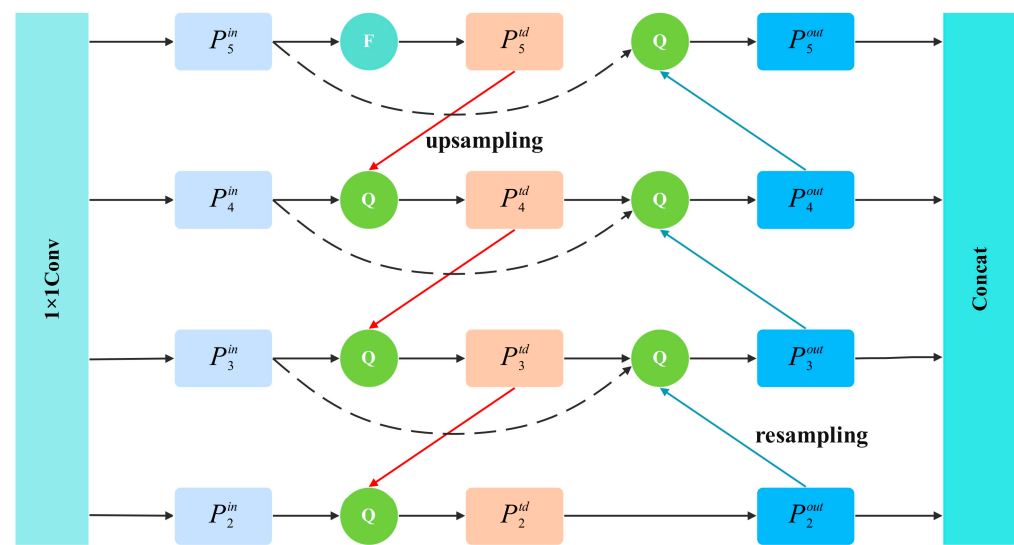


Figure 5. Structure of the BiFPN multi-scale feature fusion module.

On the basis of the top-down pathway in the standard FPN, BiFPN additionally introduces a bottom-up feature propagation pathway, allowing shallow spatial detail information and deep semantic abstraction information to interact sufficiently through bidirectional information flow. Meanwhile, BiFPN abandons the naive equal-weight summation strategy and assigns an independent learnable scalar weight to each input feature path, enabling the network to adaptively learn the relative importance of features at different scales during training. For an intermediate feature layer in the network, taking the P_4 layer as an example, the intermediate fused feature P_4^{td} is first computed in the top-down propagation stage as follows:

$$P_4^{td} = \mathcal{F} \left(\frac{\hat{w}_1 \cdot P_4^{in} + \hat{w}_2 \cdot \mathcal{R}(P_5^{td})}{\hat{w}_1 + \hat{w}_2 + \epsilon} \right) \quad (5)$$

where P_4^{in} denotes the original input feature from the backbone network, $\mathcal{R}(\cdot)$ denotes bilinear interpolation upsampling, and $\mathcal{F}(\cdot)$ denotes the feature refinement module composed of depthwise separable convolutions. $\hat{w}_i = \text{ReLU}(w_i)$ represents the non-negative learnable weight after activation constraint, and $\epsilon = 10^{-4}$ is a stability term used to prevent numerical degradation. In the subsequent bottom-up propagation stage, P_4^{td} , the original input feature P_4^{in} , and the resampled output feature from the lower layer, $\mathcal{R}(P_3^{out})$, jointly participate in three-input weighted fusion to compute the final output feature P_4^{out} :

$$P_4^{out} = \mathcal{F} \left(\frac{\hat{w}'_1 \cdot P_4^{in} + \hat{w}'_2 \cdot P_4^{td} + \hat{w}'_3 \cdot \mathcal{R}(P_3^{out})}{\hat{w}'_1 + \hat{w}'_2 + \hat{w}'_3 + \epsilon} \right) \quad (6)$$

In the above equation, the three path weights $\hat{w}'_1$, $\hat{w}'_2$ and $\hat{w}'_3$ correspond to the contribution ratios of the original input path, the top-down intermediate feature path, and the bottom-up feedback path, respectively. After normalization, they satisfy $\sum_i \hat{w}'_i / (\sum_j \hat{w}'_j + \epsilon) \in (0, 1)$, thereby ensuring the numerical stability and weight interpretability of the fusion process. The bidirectional fusion process described above is performed in parallel across the four scales from P_2 to P_5 , and two BiFPN layers ($N = 2$) are

stacked iteratively in all experiments to progressively enhance the semantic consistency of cross-scale features. Finally, the output features at all scales are upsampled to the spatial resolution of P_2 by bilinear interpolation and concatenated along the channel dimension to form the comprehensive feature map F_{fused} :

$$F_{fused} = \text{Concat}[P_2^{out}, \mathcal{R}(P_3^{out}), \mathcal{R}(P_4^{out}), \mathcal{R}(P_5^{out})] \quad (7)$$

This feature map jointly encodes spatial details and semantic information from different receptive-field levels, providing richer and more scale-balanced feature inputs for the subsequent text kernel segmentation head.

In summary, through the synergistic design of bidirectional feature flow and adaptive weighted fusion, BiFPN fundamentally overcomes the limitations of the standard FPN in dense text scenarios, namely insufficient feature interaction and fixed fusion weights. This enables the model to simultaneously preserve the semantic integrity of large-scale text regions and the boundary precision of small-scale character regions. As a result, adhesion and confusion among densely arranged text instances are effectively suppressed, and the detection accuracy and robustness are substantially improved without a significant increase in computational overhead.

3.3.3. Lightweight Detection Head Design Based on GhostModule

The original PSENet detection head uses a standard 3×3 convolution to further extract fused features, followed by a 1×1 convolution to generate prediction maps for text regions and multi-scale text kernels. This structure provides good local feature modeling capability. However, in dense text detection scenarios involving customs declaration forms, images usually contain a large number of text regions with small font sizes, narrow spacing, and high-density arrangements. Therefore, the detection head must balance fine-grained feature representation with computational efficiency. A standard 3×3 convolution performs spatial modeling and channel transformation simultaneously, which results in high parameter and computational costs. In addition, some of the generated features are redundant. To address this issue, this study replaces the standard convolution in the original detection head with GhostModule [32] and constructs a lightweight detection head, termed PSEGhostHead, as shown in Figure 6. This module first generates intrinsic features using a small number of primary convolutions, and then produces supplementary features through low-cost depthwise convolutions. In this way, it reduces computational redundancy while preserving the representation capability for text boundaries, stroke structures, and small-scale text-kernel regions, thereby improving the applicability of the detection head to complex layouts of customs declaration forms.

The improved detection head first feeds the input feature tensor $X \in \mathbb{R}^{C_{in} \times H \times W}$ into the GhostModule. Let the number of hidden channels be C_h . Following the setting used in the original GhostNet, we set the Ghost ratio to $r = 2$. Accordingly, the primary branch uses a 1×1 convolution to generate $C_h/2$ intrinsic features. Under this setting, the hidden channels are evenly divided between the primary convolution branch and the cheap depthwise convolution branch for primary semantic representation and channel information fusion, respectively. This process is formulated as follows:

$$F_p = \delta\left(\text{BN}\left(W_p^{1 \times 1} * X\right)\right), \quad F_p \in \mathbb{R}^{\frac{C_h}{2} \times H \times W} \quad (8)$$

where $*$ denotes the convolution operation, $W_p^{1 \times 1}$ represents the weights of the primary convolution, $\text{BN}(\cdot)$ denotes batch normalization, and $\delta(\cdot)$ denotes the ReLU activation function. Based on the intrinsic features, the depthwise convolution branch performs channel-wise spatial transformation to generate Ghost features. These Ghost features are

then concatenated with the intrinsic features along the channel dimension to obtain the complete hidden features:

$$F_g = \text{Concat}\left(F_p, \delta\left(\text{BN}\left(W_d^{3 \times 3} \odot F_p\right)\right)\right), \quad F_g \in \mathbb{R}^{C_h \times H \times W} \quad (9)$$

where $\odot$ denotes the depthwise convolution operation, and $W_d^{3 \times 3}$ represents the channel-wise convolution kernels. Subsequently, F_g is mapped by a 1×1 convolution to the final prediction maps, which are used to output the complete text regions and multi-scale text kernels. This process preserves the progressive scale expansion mechanism of PSENet while enhancing local features at a lower computational cost.

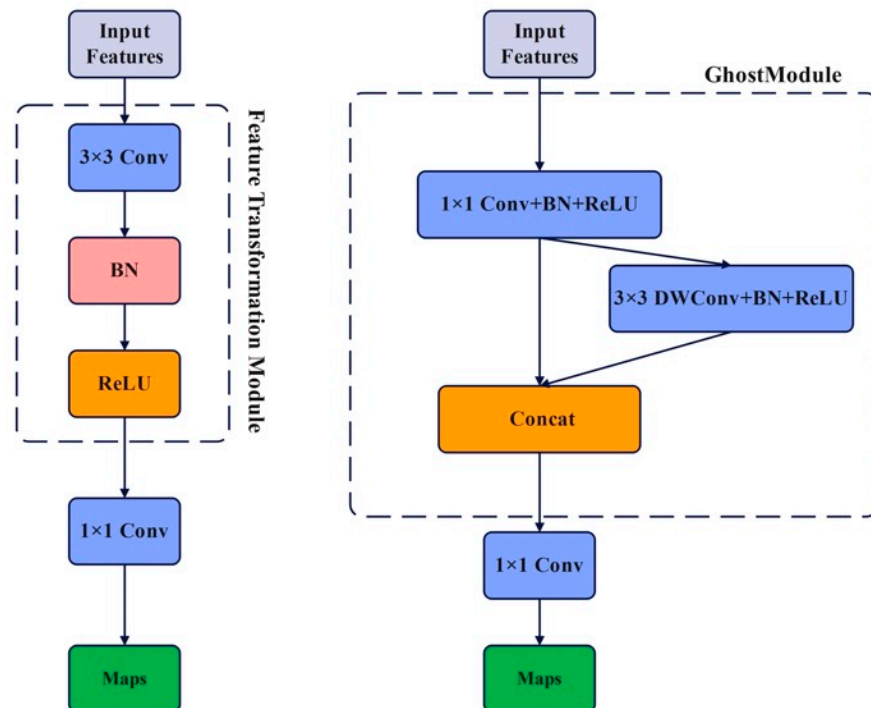


Figure 6. Architectures of the conventional PSENet detection head (**left**) and the improved PSENet detection head (**right**).

Overall, the GhostModule-based detection head optimizes the feature generation process without changing the output format of PSENet. This structure effectively reduces the computational redundancy introduced by standard convolutions while retaining the ability to perceive local structures and boundary information in dense text. For dense, small-scale, and adjacent text regions in customs declaration forms, the proposed module provides more compact and effective feature representations, thereby offering a more reliable basis for subsequent text instance separation and accurate localization.

3.4. Training Details

All experiments were performed on an A10 server. Visual Studio Code (<https://code.visualstudio.com/>) was adopted as the main integrated development environment for project development and debugging, thereby ensuring consistency and reproducibility in code implementation, model training, and experimental management. The detailed configuration of the experimental environment is listed in Table 2.

Table 2. Training environment and related parameters.

Component	Specification/Version
Operating system	Rocky Linux 8.8
Memory	502 GB
CPU	Intel Xeon Gold 6342 @ 2.80 GHz, 24 × 2 cores
GPU	NVIDIA A10
Python	3.10
PaddlePaddle	3.3.1
CUDA	11.8

During model training, the Adam optimizer was used to update the network parameters. The first-moment coefficient and second-moment coefficient were set to 0.9 and 0.999, respectively. Weight decay was implemented using L2 regularization with a coefficient of 5×10^{-4} to enhance the generalization capability of the model and reduce the risk of overfitting. For the CDTD dataset, the learning rate was fixed at 1×10^{-4} , and the model was trained for 100 epochs with a batch size of 8. For the ICDAR2015 dataset, the initial learning rate was set to 1×10^{-4} , and the model was trained for 600 epochs with a batch size of 8. The learning rate was reduced by a factor of 0.1 every 200 epochs. For both datasets, the input image size was set to 640×640 pixels. The training images were augmented using color jittering, random scaling, horizontal flipping, random rotation, and random cropping.

3.5. Evaluation Metrics

To comprehensively evaluate detection accuracy, model complexity, computational cost, storage requirements, and inference efficiency, seven metrics were adopted: Precision, Recall, F-measure, the number of parameters (Params), floating-point operations (FLOPs), Model Size, and frames per second (FPS). Precision, Recall, and F-measure quantify text detection accuracy; Params and FLOPs characterize model complexity and computational cost, respectively; Model Size reflects storage requirements; and FPS measures inference efficiency.

Text detection accuracy was evaluated at the region level. Predicted text regions were matched with ground-truth text regions using polygon intersection over union (IoU), defined as the ratio of their intersection area to their union area. A predicted text region was counted as a true positive (TP) when its IoU with an unmatched, non-ignored ground-truth region was greater than 0.5. One-to-one matching was employed, such that each predicted region and each ground-truth region could be matched once at most. An unmatched valid prediction was counted as a false positive (FP), whereas an unmatched valid ground-truth region was counted as a false negative (FN).

Ground-truth regions annotated as ignored were excluded from the evaluation. A predicted region was also excluded if more than 50% of its area overlapped an ignored ground-truth region. The numbers of TP, FP, and FN were accumulated over the entire test set. Precision, Recall, and F-measure were then calculated as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (10)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (11)$$

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (12)$$

Precision measures the reliability of the predicted text regions, Recall reflects the ability to detect ground-truth text regions, and F-measure provides their harmonic mean. Params denotes the total number of model parameters in millions (M), while FLOPs represents the computational cost of a single forward pass under a unified input resolution and is reported in G. Model Size denotes the storage space occupied by the model weight file in MB. FPS denotes the number of image frames that the model can process per second, and is used to measure the inference speed and real-time processing capability of the model. Under the same experimental environment, input image size, and inference settings, a higher FPS indicates higher inference efficiency, making the model more suitable for real-time detection tasks.

4. Results

4.1. Ablation Study on the CDTD Dataset

To validate the effectiveness of the proposed improvement modules in dense text detection for customs declaration forms, ablation experiments were conducted on the self-constructed CDTD dataset. The original PSENet-ResNet50 was used as the baseline model. Under the same training strategy and testing settings, the lightweight MobileNetV3 backbone, the BiFPN bidirectional multi-scale feature fusion module, and the PSEGhostHead lightweight detection head were successively introduced, ultimately forming the complete MBG-PSENet model. Each model was trained and evaluated in three independent runs, and the Precision, Recall, and F-measure values in Table 3 are reported as the mean $\pm$ standard deviation. The ablation results are presented in Table 3, and the performance comparisons among different models are shown in Figure 7.

Table 3. Ablation results on the CDTD dataset.

Model ID	Model	Precision/%	Recall/%	F-Measure/%	Params/M	FLOPs/G	Model Size	FPS
A	PSENet-ResNet50	99.33 $\pm$ 0.26	98.67 $\pm$ 0.24	99.00 $\pm$ 0.04	28.70	118.98	109.52 MB	19.53
B	PSENet-MobileNetV3	97.99 $\pm$ 0.48	96.99 $\pm$ 1.20	97.48 $\pm$ 0.61	1.43	11.97	5.48 MB	48.27
C	PSENet-MobileNetV3-BiFPN	99.34 $\pm$ 0.51	98.59 $\pm$ 0.16	98.96 $\pm$ 0.17	1.35	10.06	5.21 MB	39.97
D	PSENet-MobileNetV3-PSEGhostHead	98.59 $\pm$ 0.64	97.44 $\pm$ 0.42	98.01 $\pm$ 0.35	1.12	3.95	4.29 MB	55.00
E	MBG-PSENet	99.48 $\pm$ 0.26	98.30 $\pm$ 0.27	98.89 $\pm$ 0.05	1.04	2.05	4.02 MB	44.16

As shown in Table 3 and Figure 7, the original PSENet-ResNet50 achieved an F-measure of 99.00%, indicating high detection accuracy. However, its number of parameters, FLOPs, and model size reached 28.70 M, 118.98 G, and 109.52 MB, respectively, while its FPS was only 19.53. This indicates that the model has relatively high computational and storage overhead. After replacing the backbone network with MobileNetV3, the number of parameters decreased to 1.43 M, the FLOPs decreased to 11.97 G, the model size decreased to 5.48 MB, and the FPS increased to 48.27. However, the F-measure dropped to 97.48%. This suggests that the lightweight backbone can significantly improve inference efficiency, but it also weakens the feature representation capability for dense text and small-scale text to some extent.

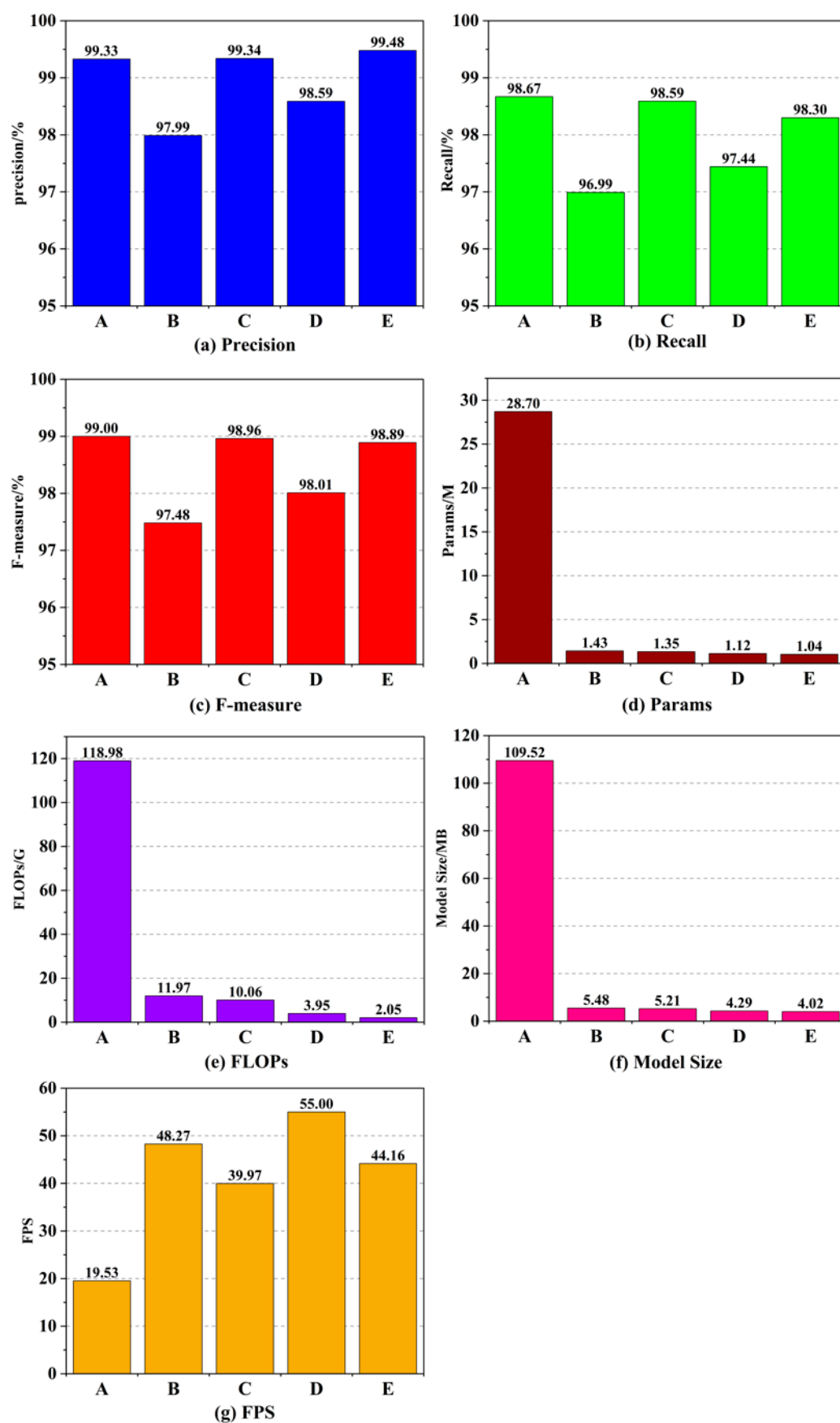


Figure 7. Performance comparison of five models on the CDTD dataset. A, PSENet-ResNet50; B, PSENet-MobileNetV3; C, PSENet-MobileNetV3-BiFPN; D, PSENet-MobileNetV3-PSEGhostHead; E, MBG-PSENet.

After BiFPN was further introduced on the basis of MobileNetV3, the F-measure increased to 98.96%, only 0.04 percentage points lower than that of the original PSENet-ResNet50. This demonstrates that bidirectional multi-scale feature fusion can effectively compensate for the accuracy loss caused by the lightweight backbone. However, despite its lower computational complexity, BiFPN involves repeated resampling, learnable weighted fusion, and cross-scale feature transfer, which likely increase memory-access and kernel-launch overhead, limit GPU utilization, and consequently reduced the FPS from 48.27 to 39.97. When PSEGhostHead was adopted, the number of parameters was further reduced to 1.12 M, the FLOPs decreased to 3.95 G, the model size decreased to 4.29 MB, and the FPS increased to 55.00. These results verify the effectiveness of the lightweight detection head in reducing redundant computation and improving inference efficiency.

Figure 8 presents the inference results of different models. As shown in Figure 8b, PSENet-ResNet50 achieved generally good detection performance, but a small number of false detections still occurred. In Figure 8c, PSENet-MobileNetV3 showed more obvious missed detections, indicating that using only the lightweight backbone reduces the detection capability for dense text and small-scale text. In contrast, MBG-PSENet, as shown in Figure 8d, produced fewer missed detections and more complete detection results. This indicates that the combination of BiFPN and PSEGhostHead can enhance text-region localization while reducing model complexity.



Figure 8. Comparison of detection results among different models. Sensitive information in the customs declaration image has been masked with black rectangles for privacy protection, and the cyan boxes indicate the text regions predicted by each model.

After integrating the three improvement modules, MBG-PSENet contained only 1.04 M parameters, required only 2.05 G FLOPs, and had a model size of 4.02 MB, representing reductions of approximately 96.38%, 98.28%, and 96.33%, respectively, compared with the original PSENet-ResNet50. Meanwhile, the FPS of MBG-PSENet increased to 44.16, and its F-measure reached 98.89%, which was only 0.11 percentage points lower than that of the original PSENet-ResNet50. These results demonstrate that MBG-PSENet substantially reduces model complexity, computational cost, and storage overhead while maintaining

high detection accuracy, achieving an effective balance between accuracy and efficiency in dense text detection for customs declaration forms.

4.2. Ablation Study on the ICDAR2015 Dataset

To further evaluate the effectiveness and generalization capability of the proposed improvement modules in general scene text detection, ablation experiments were conducted on the public benchmark dataset ICDAR2015. This experiment was designed to verify the applicability of the lightweight MobileNetV3 backbone, the BiFPN multi-scale feature fusion module, and the PSEGhostHead lightweight detection head to unstructured scene text detection. The experimental results are presented in Table 4, and the performance comparisons among different models are shown in Figure 9.

Table 4. Ablation results on the ICDAR2015 dataset.

Model ID	Model	Precision/%	Recall/%	F-Measure/%	Params/M	FLOPs/G	Model Size	FPS
A	PSENet-ResNet50	85.50	78.91	82.07	28.70	118.98	109.52 MB	18.34
B	PSENet-MobileNetV3	80.78	70.32	75.19	1.43	11.97	5.48 MB	42.47
C	PSENet-MobileNetV3-BiFPN	82.49	71.45	76.57	1.35	10.06	5.21 MB	39.71
D	PSENet-MobileNetV3-PSEGhostHead	80.86	70.97	75.59	1.12	3.95	4.29 MB	52.93
E	MBG-PSENet	83.30	71.30	76.84	1.04	2.05	4.02 MB	41.44

As shown in Table 4 and Figure 9, the original PSENet-ResNet50 achieved an F-measure of 82.07%. However, its number of parameters, FLOPs, and model size reached 28.70 M, 118.98 G, and 109.52 MB, respectively, while its FPS was only 18.34. This indicates that although the model has relatively high detection accuracy, it also involves considerable computational and storage overhead. After replacing the backbone network with MobileNetV3, the number of parameters decreased to 1.43 M, the FLOPs decreased to 11.97 G, the model size decreased to 5.48 MB, and the FPS increased to 42.47, but the F-measure dropped to 75.19%. This suggests that the lightweight backbone can significantly improve inference efficiency, but it weakens the feature representation capability in scenarios with complex backgrounds, multi-oriented text, and large scale variations.

After BiFPN was further introduced, the F-measure increased to 76.57%, and both Precision and Recall were improved. The number of parameters and FLOPs were 1.35 M and 10.06 G, respectively, indicating that BiFPN enhanced multi-scale feature fusion while maintaining a lightweight computational cost. This indicates that bidirectional multi-scale feature fusion can effectively alleviate the accuracy loss caused by the lightweight backbone. When PSEGhostHead was adopted, the number of parameters was further reduced to 1.12 M, the FLOPs decreased to 3.95 G, the model size decreased to 4.29 MB, and the FPS increased to 52.93, verifying the effectiveness of the lightweight detection head in reducing redundant computation. After integrating the three improvement modules, MBG-PSENet achieved an F-measure of 76.84%, with only 1.04 M parameters, 2.05 G FLOPs, a model size of 4.02 MB, and an FPS of 41.44. These results demonstrate that MBG-PSENet substantially reduces model complexity, computational cost, and storage overhead while maintaining relatively good detection performance on the ICDAR2015 dataset, reflecting its lightweight advantage and certain generalization capability.

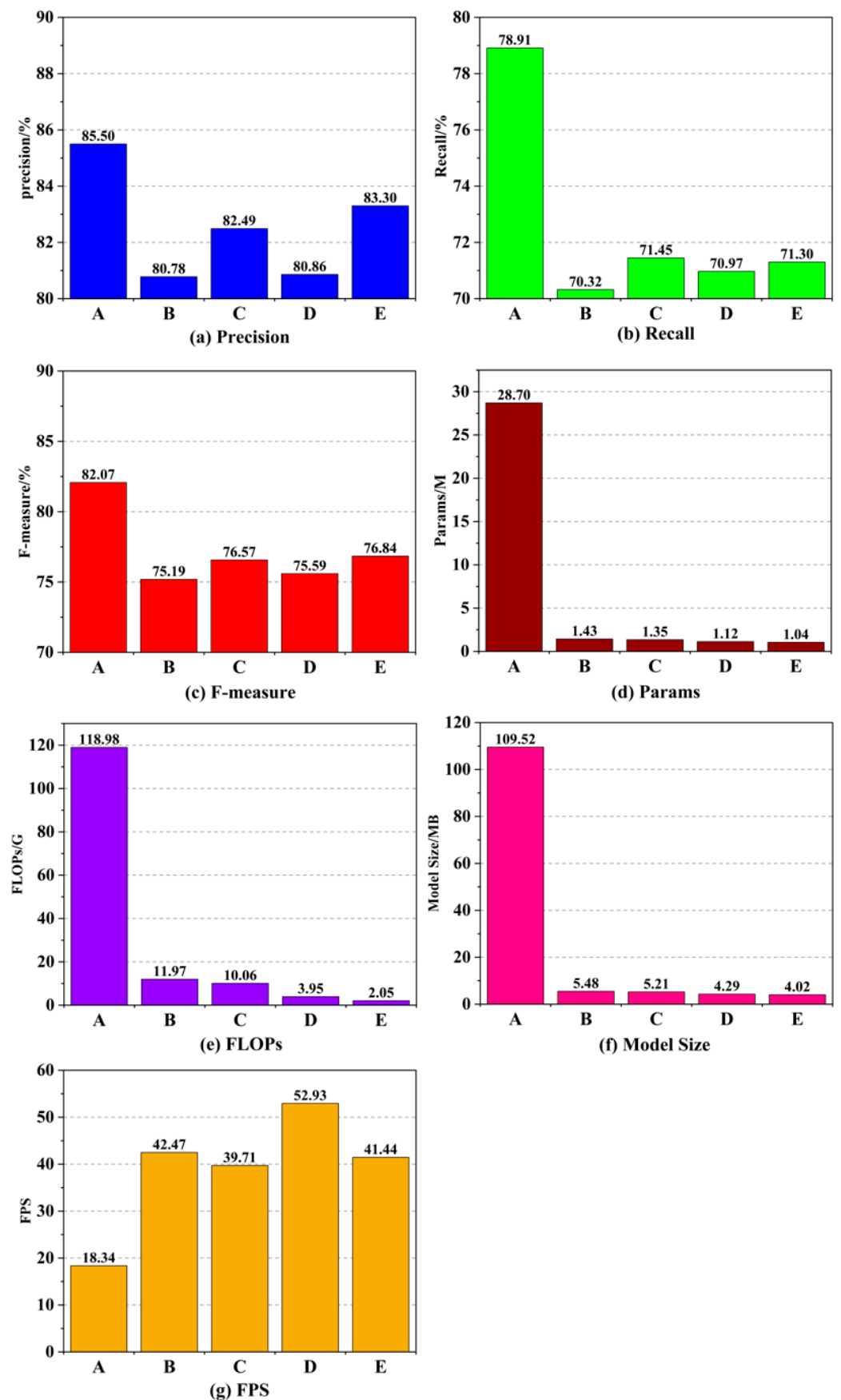


Figure 9. Performance comparison of five models on the ICDAR2015 dataset. A, PSENet-ResNet50; B, PSENet-MobileNetV3; C, PSENet-MobileNetV3-BiFPN; D, PSENet-MobileNetV3-PSEGhostHead; E, MBG-PSENet.

4.3. Comparative Performance Evaluation on the CDTD Dataset

To further evaluate the overall performance of the proposed MBG-PSENet for dense text detection in customs declaration forms, comparative experiments were conducted on the self-constructed CDTD dataset using several representative text detection models, including DBNet, DBNet++, EAST, SAST, and the original PSENet. All models in this comparison were implemented and evaluated under the same hardware environment, software environment, and evaluation protocol to ensure a fair comparison. The quantitative results of each model are reported in Table 5, and the performance comparison among different models is shown in Figure 10.

Table 5. Performance Comparison of Different Models on the CDTD Dataset.

Model ID	Model	Precision/%	Recall/%	F-Measure/%	Params/M	FLOPs/G	Model Size	FPS
A	DBNet [23]	99.20	98.95	99.08	25.48	49.26	97.24 MB	23.32
B	PSENet-MobileNetV3	97.99	96.99	97.48	1.43	11.97	5.48 MB	48.27
C	DBNet++ [24]	99.45	99.05	99.25	26.48	40.34	101.05 MB	19.72
D	EAST-MobileNetV3 [13]	96.27	96.73	96.50	1.67	7.45	6.41 MB	61.07
E	SAST [33]	98.75	95.38	97.04	47.83	95.57	182.50 MB	17.84
F	PSENet-ResNet50	99.33	98.67	99.00	28.70	118.98	109.52 MB	19.53
G	MBG-PSENet	99.48	98.30	98.89	1.04	2.05	4.02 MB	44.16

The experimental results show that MBG-PSENet achieved the highest Precision among all compared models, reaching 99.48%. This indicates that the proposed model can effectively reduce false detections caused by complex interference factors, such as table lines, stamps, and scanning noise. Its Recall and F-measure were 98.30% and 98.89%, respectively. Although the F-measure of MBG-PSENet was slightly lower than those of DBNet++, DBNet, and PSENet-ResNet50, the performance gaps were small, with differences of only 0.36, 0.19, and 0.11 percentage points, respectively. Meanwhile, its detection performance was clearly superior to that of lightweight detection models such as PSENet-MobileNetV3 and EAST-MobileNetV3.

In terms of model complexity and computational efficiency, MBG-PSENet demonstrated more pronounced advantages. The model contained only 1.04 M parameters, required only 2.05 G FLOPs, and had a model size of 4.02 MB, all of which were the lowest among the compared methods. Overall, while maintaining high text detection accuracy, MBG-PSENet significantly reduced the model parameters, computational cost, and storage overhead, achieving a favorable balance among detection accuracy, model size, and computational efficiency. These results indicate that MBG-PSENet is more suitable for deployment in resource-constrained customs declaration OCR systems.

4.4. Generalization Performance Evaluation on the ICDAR2015 Dataset

To further evaluate the generalization capability of MBG-PSENet for text detection in unstructured natural scenes, comparative experiments were conducted on the public ICDAR2015 dataset. The proposed method was compared with several representative text detection models, and the experimental results are presented in Table 6.

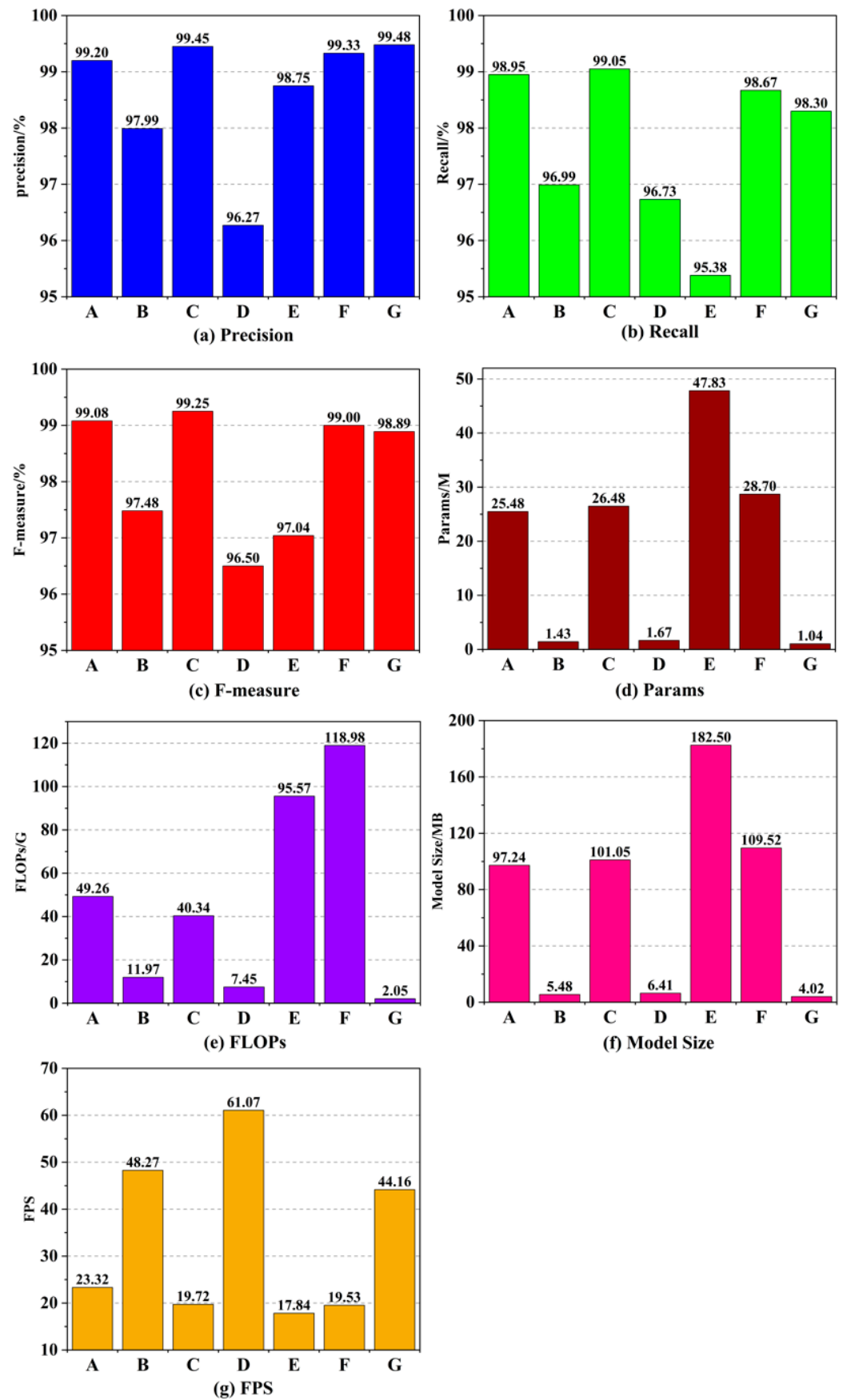


Figure 10. Performance comparison of different models on the CDTD dataset. A, DBNet; B, PSENet-MobileNetV3; C, DBNet++; D, EAST-MobileNetV3; E, SAST; F, PSENet-ResNet50; G, the proposed MBG-PSENet.

Table 6. Comparison of different models on the ICDAR2015 dataset.

Model	Precision/%	Recall/%	F-Measure/%	Params/M	FLOPs/G	Model Size	FPS
CTPN [12]	74.20	51.60	60.90	-	-	-	-
EAST (VGG16 + RBOX) [13]	80.46	72.75	76.41	-	-	-	-
DBNet-ResNet50	88.50	77.80	82.81	25.48	49.26	97.24 MB	22.40
DBNet-MobileNetV3	74.99	72.17	73.55	1.86	11.38	7.11 MB	27.75
PSENet-MobileNetV3	80.78	70.32	75.19	1.43	11.97	5.48 MB	42.47
PSENet-ResNet50	85.50	78.91	82.07	28.70	118.98	109.52 MB	18.34
MBG-PSENet	83.30	71.30	76.84	1.04	2.05	4.02 MB	41.44

As shown in Table 6, MBG-PSENet achieved a Precision of 83.30%, a Recall of 71.30%, and an F-measure of 76.84% on the ICDAR2015 dataset. Its overall performance was superior to that of CTPN, DBNet-MobileNetV3, and PSENet-MobileNetV3, with F-measure improvements of 15.94, 3.29, and 1.65 percentage points, respectively. In addition, the F-measure of MBG-PSENet was 0.43 percentage points higher than that of EAST (VGG16 + RBOX), indicating that the proposed multi-scale feature fusion strategy and lightweight detection head can maintain relatively stable text-region representation capability in natural scene images.

Although the F-measure of MBG-PSENet was lower than those of DBNet-ResNet50 and PSENet-ResNet50, with gaps of 5.97 and 5.23 percentage points, respectively, this difference can be mainly attributed to the limited capacity of the lightweight backbone network in fine-grained semantic modeling and complex contextual representation. Nevertheless, MBG-PSENet contains only 1.04 M parameters, requires only 2.05 G FLOPs, and has a model size of 4.02 MB, which are substantially lower than those of the ResNet50-based detection models. Meanwhile, MBG-PSENet achieved an inference speed of 41.44 FPS, which is much higher than those of DBNet-ResNet50 and PSENet-ResNet50. Overall, MBG-PSENet substantially reduces model complexity, computational cost, and storage overhead while maintaining relatively stable detection accuracy, thereby demonstrating its lightweight advantages and a certain degree of cross-scene generalization capability in general scene text detection tasks.

5. Discussion

The experimental results indicate that the proposed architectural modifications effectively adapt PSENet to dense text detection in customs declaration forms, particularly by improving lightweight deployment while preserving reliable localization performance. Compared with the original PSENet-ResNet50, the proposed model substantially reduces the number of parameters and the model size while maintaining a high F-measure on the CDTD dataset. This indicates that the combination of MobileNetV3, BiFPN, and PSEGhost-Head provides an effective complementarity between lightweight network design and multi-scale text feature representation. Specifically, BiFPN partly compensates for the loss of feature representation capacity caused by the lightweight backbone, whereas PSEGhost-Head further reduces redundant computation in the detection head. These characteristics make the proposed model more suitable for customs declaration OCR applications with strict requirements for storage capacity, inference speed, and deployment cost. The results on the ICDAR2015 dataset also suggest that MBG-PSENet has a certain degree of generalization ability in unstructured natural scene text detection. However, its main advantage lies in lightweight design and computational efficiency, rather than consistently outperforming heavyweight detection models in all complex scenarios.

This study still has several limitations. First, although the CDTD dataset was collected from real customs declaration scenarios, its data scale, template diversity, acquisition devices, image quality, and layout variations remain limited. Therefore, it may not fully cover

the complex conditions encountered in practical customs document processing. Second, this study mainly evaluates model performance from the perspective of text detection. The influence of detection errors on subsequent text recognition, field matching, layout understanding, and key information extraction has not yet been systematically investigated. Therefore, the actual benefit of the proposed model in an end-to-end customs declaration understanding system requires further validation. Third, the Recall of MBG-PSENet on the ICDAR2015 dataset was lower than that of some ResNet50-based heavyweight models, indicating that the lightweight backbone still has limitations in fine-grained feature representation for complex backgrounds, small-scale text, blurred text, occluded text, and arbitrarily oriented text. Future work can be carried out in the following directions. First, more customs declaration forms and similar structured documents from diverse sources, templates, and degradation types should be collected to improve cross-scenario robustness. Since many structured business documents, such as invoices, logistics forms, bills of lading, and receipts, also contain dense text, table lines, and multi-scale fields, MBG-PSENet may be extended to these similar document analysis scenarios after domain-specific fine-tuning and evaluation. Second, while preserving the lightweight advantage of the model, hard example mining, boundary refinement, lightweight attention mechanisms, or more effective multi-scale contextual modeling strategies can be introduced to improve the recall of low-quality and small-scale text regions. Third, text detection should be jointly evaluated with text recognition, layout analysis, and key information extraction, so that the practical value of the model can be further verified using end-to-end business-oriented metrics.

6. Conclusions

This study proposes MBG-PSENet, a lightweight text detection model for customs declaration images. The model combines a MobileNetV3 backbone, a BiFPN-based multi-scale feature fusion module, and a GhostModule-based PSEGhostHead. This design preserves the progressive scale expansion mechanism of PSENet while reducing redundant computation in both feature extraction and prediction. It therefore provides a compact framework for dense text detection in structured document images.

The experimental results demonstrate that MBG-PSENet achieves a favorable balance between detection accuracy and deployment efficiency. On the CDTD dataset, it achieved a 99.48% Precision, 98.30% Recall, and 98.89% F-measure, with only 1.04 M parameters, 2.05 G FLOPs, a model size of 4.02 MB, and an inference speed of 44.16 FPS. Compared with the original PSENet-ResNet50, MBG-PSENet substantially reduces parameters, computation, and storage costs while maintaining comparable detection accuracy. The results on ICDAR2015 further indicate that the proposed model retains a certain degree of generalization ability in natural scene text detection.

The theoretical value of this study lies in adapting the PSENet framework to dense document text detection through coordinated lightweight design. MobileNetV3 reduces backbone complexity, BiFPN strengthens bidirectional multi-scale feature interaction, and PSEGhostHead reduces redundant feature generation in the detection head. Their combination shows that lightweight feature extraction, multi-scale semantic compensation, and low-cost prediction can be jointly used to improve the efficiency of segmentation-based text detection.

From a practical perspective, MBG-PSENet is suitable for customs declaration OCR systems that require small model size, low computational cost, and fast inference. These characteristics are important for document-processing scenarios with limited computing resources, high processing volume, or deployment on edge devices. Therefore, the proposed method has practical potential for automatic customs declaration processing and can also provide a reference for text detection in other structured business documents. Future work

will focus on deployment-oriented optimization, broader structured-document validation, and integration with downstream text recognition and information extraction modules.

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

Funding: This research was funded by the Key Research and Development Program of Xinjiang Uygur Autonomous Region, under the project “Research on Key Technologies of Digital Twin and Intelligent Management and Control for Smart Warehousing in Cross-Border Logistics”, grant number 2025B04045.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available from the corresponding author upon reasonable request. Researchers requesting access for academic research should provide a clear statement describing the intended research use.

Acknowledgments: The authors would like to express their sincere gratitude to all members of the research team for their contributions to the data preparation, experimental validation, and result analysis. The authors also gratefully acknowledge Horgos Jinyi International Trade (Group) Co., Ltd. for providing the dataset used in this study.

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

References

1. Wang, X.F.; He, Z.H.; Wang, K.; Wang, Y.F.; Zou, L.; Wu, Z.Z. A survey of text detection and recognition algorithms based on deep learning technology. *Neurocomputing* **2023**, *556*, 126702.
2. Sudana, O.; Wirdiani, A.; Putra, A.D.W. Fine-tuned LayoutLMv3 for Indonesian receipts extraction. *Bull. Electr. Eng. Inform.* **2026**, *15*, 1127–1138. [[CrossRef](#)]
3. Wei, H.; Liu, C.; Chen, J.; Wang, J.; Kong, L.; Xu, Y.; Ge, Z.; Zhao, L.; Sun, J.; Peng, Y.; et al. General OCR theory: Towards OCR-2.0 via a unified end-to-end model. *arXiv* **2024**, arXiv:2409.01704.
4. Xu, Y.; Li, M.; Cui, L.; Huang, S.; Wei, F.; Zhou, M. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1192–1200.
5. Ye, M.; Zhang, J.; Zhao, S.; Liu, J.; Du, B.; Tao, D. Dptext-detr: Towards better scene text detection with dynamic points in transformer. In *AAAI’23/IAAI’23/EAAI’23: Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2023; Volume 37, pp. 3241–3249.
6. Huang, L.; Yang, W. GP-PSENet: A group-related dilated and a parallel extensional dilation-wise residual encoder for scene text detection. *Neural Comput. Appl.* **2025**, *37*, 7159–7181. [[CrossRef](#)]
7. Wang, G.; Wei, S.; Yang, D.; Guo, A. Natural Scene Text Detection Algorithm via Internal Feature Enhancement and Adaptive Cross Fusion. *IEEE Access* **2025**, *13*, 153159–153170. [[CrossRef](#)]
8. Chen, Z. (HTBNet)Arbitrary Shape Scene Text Detection with Binarization of Hyperbolic Tangent and Cross-Entropy. *Entropy* **2024**, *26*, 560. [[CrossRef](#)] [[PubMed](#)]
9. Wang, Y.; Mamat, H.; Xu, X.; Aysa, A.; Ubul, K. Scene Uyghur Text Detection Based on Fine-Grained Feature Representation. *Sensors* **2022**, *22*, 4372. [[CrossRef](#)] [[PubMed](#)]
10. Wang, W.; Xie, E.; Li, X.; Hou, W.; Lu, T.; Yu, G.; Shao, S. Shape robust text detection with progressive scale expansion network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 15–20 June 2019; pp. 9336–9345.
11. Naiemi, F.; Ghods, V.; Khalesi, H. Scene text detection and recognition: A survey. *Multimed. Tools Appl.* **2022**, *81*, 20255–20290. [[CrossRef](#)]

12. Tian, Z.; Huang, W.; He, T.; He, P.; Qiao, Y. Detecting text in natural image with connectionist text proposal network. In *European Conference on Computer Vision*; Springer International Publishing: Cham, Switzerland, 2016; pp. 56–72.
13. Zhou, X.; Yao, C.; Wen, H.; Wang, Y.; Zhou, S.; He, W.; Liang, J. East: An efficient and accurate scene text detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2017; pp. 5551–5560.
14. Liao, M.; Shi, B.; Bai, X. Textboxes++: A single-shot oriented scene text detector. *IEEE Trans. Image Process.* **2018**, *27*, 3676–3690. [[CrossRef](#)] [[PubMed](#)]
15. Liao, M.; Zhu, Z.; Shi, B.; Xia, G.S.; Bai, X. Rotation-sensitive regression for oriented scene text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2018; pp. 5909–5918.
16. Zhu, Y.; Chen, J.; Liang, L.; Kuang, Z.; Jin, L.; Zhang, W. Fourier contour embedding for arbitrary-shaped text detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2021; pp. 3123–3131.
17. Deng, D.; Liu, H.; Li, X.; Cai, D. Pixellink: Detecting scene text via instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2018; Volume 32.
18. Xu, Y.; Wang, Y.; Zhou, W.; Wang, Y.; Yang, Z.; Bai, X. Textfield: Learning a deep direction field for irregular scene text detection. *IEEE Trans. Image Process.* **2019**, *28*, 5566–5579. [[CrossRef](#)] [[PubMed](#)]
19. Long, S.; Ruan, J.; Zhang, W.; He, X.; Wu, W.; Yao, C. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 20–36.
20. Baek, Y.; Lee, B.; Han, D.; Yun, S.; Lee, H. Character region awareness for text detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2019; pp. 9365–9374.
21. Wang, W.; Xie, E.; Song, X.; Zang, Y.; Wang, W.; Lu, T.; Yu, G.; Shen, C. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*; IEEE: New York, NY, USA, 2019; pp. 8440–8449.
22. Wang, W.; Xie, E.; Li, X.; Liu, X.; Liang, D.; Yang, Z.; Lu, T.; Shen, C. Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 5349–5367.
23. Liao, M.; Wan, Z.; Yao, C.; Chen, K.; Bai, X. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2020; Volume 34, pp. 11474–11481.
24. Liao, M.; Zou, Z.; Wan, Z.; Yao, C.; Bai, X. Real-time scene text detection with differentiable binarization and adaptive scale fusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 919–931. [[CrossRef](#)] [[PubMed](#)]
25. Sheng, T.; Chen, J.; Lian, Z. Centripetaltext: An efficient text instance representation for scene text detection. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 335–346.
26. Karatzas, D.; Gomez-Bigorda, L.; Nicolaou, A.; Ghosh, S.K.; Bagdanov, A.; Iwamura, M.; Matas, J.; Neumann, L.; Chandrasekhar, V.R.; Lu, S.; et al. ICDAR 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*; IEEE: New York, NY, USA, 2015; pp. 1156–1160.
27. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2017; pp. 2117–2125.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2016; pp. 770–778.
29. Howard, A.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*; IEEE: New York, NY, USA, 2019; pp. 1314–1324.
30. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2018; pp. 7132–7141.
31. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2020; pp. 10781–10790.
32. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2020; pp. 1580–1589.
33. Wang, P.; Zhang, C.; Qi, F.; Huang, Z.; En, M.; Han, J.; Liu, J.; Ding, E.; Shi, G. A single-shot arbitrarily-shaped text detector based on context attended multi-task learning. In *Proceedings of the 27th ACM International Conference on Multimedia*; Association for Computing Machinery: New York, NY, USA, 2019; pp. 1277–1285.

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