# Context-Aware Edge-Cloud Collaborative Scene Text Recognition

Puning Zhang[1,2,3], Changfeng Liu[1,2,3], Honggang Wang[4], Dapeng Wu[1,2,3], Ruyan Wang[1,2,3], Hong Zou[1,2,3]
[1]*School of Communications and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China.*
[2]*Advanced Network and Intelligent Connection Technology Key Laboratory of Chongqing Education Commission of China.*
[3]*Chongqing Key Laboratory of Ubiquitous Sensing and Networking, Chongqing, China.*
[4]*Katz School of Science and Health, Yeshiva University, USA.*
Email: zhangpn@cqupt.edu.cn, s210131150@stu.cqupt.edu.cn, hwang1@umassd.edu, wudp@cqupt.edu.cn, wangry@cqupt.edu.cn, zouhong@cqupt.edu.cn

***Abstract*—Scene text recognition can extract key information in the image to enable the machine to understand the semantics contained in the image, which can also play a vital role in tasks such as video understanding or video clip positioning. There are multiple irregular texts in various scene images. How to identify such irregular texts to better understand the semantics contained in the images is a tough challenge. Most of the existing scene text detection methods adopt a unified model to detect scene text indiscriminately when dealing with scene text of arbitrary shape, which results in the high model complexity and large detection delay. Facing the above problems, a context-aware edge-cloud collaborative scene text recognition method is proposed. First, an edge-cloud collaborative scene text detection architecture is designed to take advantages of the computing resource characteristics of the cloud layer and the edge layer for collaborative scene text detection. Second, a multi-category scene text context awareness method is proposed, and the attention mechanism is introduced to perceive the scene where it is located to realize the rapid classification of multi-category scene text. Furthermore, an adaptive multi-category scene text detection method is devised, and the optimal detection strategy is determined according to the classification results to reduce the computational cost of model detection. Validation results show that our method compared with traditional methods can effectively reduce 19% of the detection delay while ensuring recognition accuracy in comparison with existing methods.**

***Index Terms*—scene text recognition, edge-cloud collaboration, context awareness, scene text classification.**

## I. Introduction

The text has always been a key information dissemination channel in human society. Nowadays, the detection of scene text has received widespread attentions. Existing methods can be summarized into two categories, regular scene text detection methods and irregular scene text detection methods. For the former, Feng et al. [1], proved that previous methods only considered visual appearance features, which were easily affected by changes in viewing angle and lighting. Therefore, they introduced character semantics to solve this problem. J. Tang et al. [2] introduced an idea of representing the overall features with a few features, modeling the relationship between the sampled features, grouping the sampled features, and since each feature group corresponds to a text instance, its bounding box could be correspondingly obtained without any post-processing operations.For the latter, Qiao, Liang et al. [3], studied the detection and recognition of arbitrarily shaped scene text, and proposed a method based on angle regression and boundary regression. Liu et al. [4] adopted the Bezier curve method to detect arbitrarily shaped scene text. However, the scene text encountered in real-life situations often includes both regular and irregular text. None of the existing literature considers scenarios involving both of the two text recognition tasks. Some methods for curved scene text detection and recognition treat non-curved scene text as curved text, which partially addresses the challenge of mixed scene text detection and recognition. Nevertheless, this idea increases model complexity, resulting in longer overall text detection and recognition delay.

Regarding the issue above, a context-aware edge-cloud collaborative scene text recognition method is proposed. The specific contributions of this paper are listed as follows:

- A scene text detection framework based on edge-cloud collaboration is designed. It fully considers the resource characteristics of the edge layer and cloud layer, and combine their computing resources to perform collaborative scene text detection.
- A context-aware classification method for multi-category scene text is proposed. Existing methods fail to achieve fast and accurate detection of multi-category scene text.

Dapeng Wu is corresponding author.

This research was supported in part by supported by National Natural Science Foundation of China (61901071, U20A20157), Natural Science Foundation of Chongqing, China (cstc2020jcyj-zdxmX0024, CSTB2022NSCQ-MSX0600), University Innovation Research Group of Chongqing (CXQT20017), Program for Innovation Team Building at Institutions of Higher Education in Chongqing (CXTDX201601020), Science and Technology Research Program of Chongqing Municipal Education Commission (KJQN202000626), Youth Innovation Group Support Program of ICE Discipline of CQUPT (SCIE-QN-2022-04) and Chongqing Municipal Technology Innovation and Application Development Special Key Project (cstc2020jscx-dxwtBX0053).

Considering that the scene text has a strong correlation with the semantics of scene image, this paper introduces a context-aware method to rapidly classify the scene text.

- An adaptive multi-category scene text recognition method is presented. In order to reduce the delay while ensuring the detection accuracy, this paper proposes a lightweight adaptive multi-category scene text detection method. According to different types of scene text, detection models with corresponding complexity are selected to reduce the time delay consumed by detection.

The following contents are organized as follows. Section II designs the recognition architecture. Section III proposes the context-aware classification method. Section IV presents the adaptive multi-category scene text detection method. Section V verifies the proposed methods. Section VI concludes this paper.

## II. Edge-Cloud Collaborative Scene Text Recognition Architecture

Traditional scene text detection methods often struggle to balance the high accuracy and low latency. Existing algorithms mainly focus on improving the structure of recognition architecture, neglecting to design a reasonable detection architecture based on the characteristics of the actual scene. However, appropriate architecture can significantly enhance the detection efficiency. Therefore, we design an Edge-Cloud Collaborative Scene Text Detection Architecture, which consists of both cloud and edge layers according to the characteristics of equipment resources in the recognition scene, as illustrated in Fig. 1.

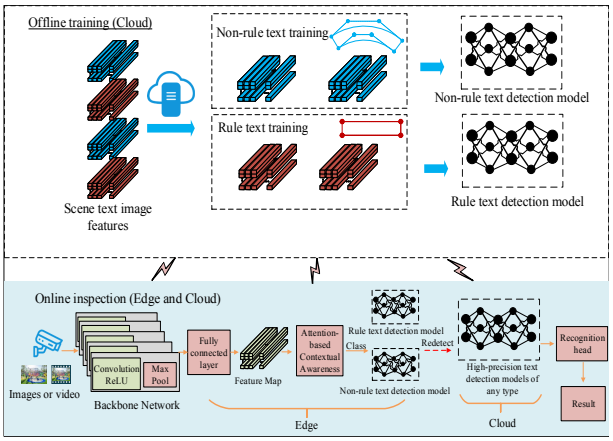


Fig. 1. Edge-Cloud collaborative scene text classification multi-strategy detection framework.

Compared to traditional architectures, the proposed framework integrates edge computing and cloud computing to enhance the efficiency and accuracy. In this architecture, the cloud primarily handles the training of text detection models, generating various types of detection models, which are then offloaded to the edge for detection. The edge is responsible for feature extraction and context awareness from input scene images, determining the category of scene text, and adaptively detecting multi-category scene text. Within this architecture, due to the strong computing power of the cloud, larger data sets and more complex models can be adopted for enhancing the accuracy of scene text detection. Moreover, since the text detection models are pre-trained in the cloud, it can be directly offloaded to the edge after training, thereby reducing the consumptions of computing and storage resources and improving detection efficiency.

Furthermore, the edge employs an adaptive scene text detection approach, utilizing varying detection strategies based on the features of text within the scene. Once the text features are classified via context aware module, the corresponding recognition module can perform detection tasks. Otherwise, the scene images are sent to the cloud for detection using a high-precision, arbitrary-type text detection model. The proposed framework is adaptable for text detection and recognition across diverse scenarios, ensuring detection accuracy while minimizing recognition delay to fulfill real-time recognition task requirements. The specific processes are shown in Fig. 2.

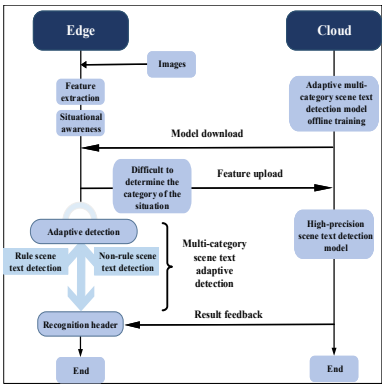


Fig. 2. Collaborative context-aware adaptive detection workflow.

When a picture or video is input, the edge will perform feature extraction on the input data, and adopt the extracted features to perform context-aware operations, then adaptively select the detection model downloaded from the cloud according to the result of context aware module for fast identification , and finally feedback the recognition result. When the input image is too complex to mine its contextual information, the edge will upload the picture to the cloud, and the high-precision scene text detection model deployed on the cloud will detect it and finally return the recognition result. In general, the scene text detection architecture combines edge computing and cloud computing and effectively leverages the computational power of the cloud and the real-time capabilities of the edge, so as to enhance the efficiency and accuracy of scene text detection.

## III. Multi-Category Scene Text Context-Aware Classification

Based on the idea that the shape of scene text often has a strong correlation with the semantics of scene image, we propose a multi-category scene text context-aware classification

method. It establishes a connection between certain landmarks in the scene and the scene text, and design a multi-category scene text context-aware classification model. When scene text image are input, the context-aware model will not only focus on the scene text, but will consider the global context features of the image to determine the type of scene text.

### A. Scene Text Analysis

In this paper, two irregularly shaped scene text datasets, CTW1500 [5] and Total text [6], are analyzed, and it is found that 95% of the CTW1500 datasets contain both irregular scene text and cartoon signs or arc markers. Such images make up 94% of the Total text dataset. The scene text dataset of regular shape of ReCTS [7] is analyzed, and we also find that the images of regular-shaped markers or buildings around the regular text account for 98% in the ReCTS dataset. Based on this, we can deduce that the shape category of the scene text is directly related to the background of its scene image. some examples are shown in Fig. 3.


Fig. 3. Examples of text images of different datasets and real-life surrounding scenes.

### B. Context-Aware Model

In this module, inspired by Transformer [8] and Swin Transformer [9], this paper introduces a multi-level attention mechanism to model images, and designs an attention-based multi-category scene text context-aware classification module. The main steps of this module can be defined as finding the attention of the image based on the two parts of the window and the pyramid.

According to the window-based attention mechanism, the image is divided into several fixed-size windows. The self-attention mechanism is then employed to calculate the attention scores between each window and other windows. This process yields attention outputs for each window. The specific steps are as follows:

$$Q_i = X_i W_q, K_j = X_j W_k \tag{1}$$

$$A_{i,j} = softmax\left(\frac{Q_i K_j^\top}{\sqrt{d}}\right) \tag{2}$$

$$Z_i = \frac{1}{\sqrt{d}} \sum_j A_{i,j} V_j \tag{3}$$

Where $Q_i$ is the query vector of window $i$, $K_j$ is the key vector of all windows, $Q_i$ and $K_j$ are obtained by linear transformations $W_q$ and $W_v$, $d$ is the feature dimension, $A_{ij}$ is the attention score between window $i$ and other windows, $V_j$ is the value vector, and $Z_i$ is the attention output of each window.

In the feature pyramid attention mechanism, different levels of features are combined through downsampling and convolution operations. The attention mechanism is then utilized to characterize spatial relationships at different scales within the image, which are defined as Equation 4 and Equation 5.

Specifically, for the adjacent two levels, we need to first obtain the query vector $Q_{i,j}$ of the current layer by convoluting and downsampling the feature $U^{l-1}$ of the previous layer, and then apply the window-based attention mechanism to obtain the attention score $A_{i,j}$ between window $i$ and all window $j$. Finally, we apply the obtained attention weights to feature $U^{l-1}$ of the previous layer, and combine them together to obtain the attention output $Z^l$ of the current layer.

$$A_{i,j} = softmax\left(\frac{Q_{i,j} K_j^\top}{\sqrt{d}}\right) \tag{4}$$

$$Z^l = \frac{1}{\sqrt{d}} \sum_{j=1}^{nl} \left( \sum_{i \in \Omega_j} A_{i,j} D\left(U_{\phi_i}^{l-1}\right) \right) W_v \tag{5}$$

Among them, $Z^l$ is the attention output of the current layer, $D$ represents the downsampling operation, $\Omega_j$ denotes the collection of windows around the $j$-th window, and $\phi_i$ means the spatial position corresponding to the feature with index $i$. $W_v$ is the model parameter that needs to be learned. By repeating the above operations, we can transfer information between different levels and combine visual features at different scales to obtain better image feature representation.

In the process of context-aware RoI, the input is an image with a size of $H \times W \times C$, the target region of interest RoI, the attention query vector $Q$, the attention key-value pair $(K, V)$, and the attention output vector $A$. The procedure performed by the module is defined as follows: first, we need to encode the image, which is to utilize the encoder to perform feature extraction and encoding on the input scene image to obtain a feature tensor $X$. Here we adopt the ResNet encoder to encode the image. The second step is query vector calculation, which is to calculate the query vector $Q$ of the desired region of interest RoI, so as to match the key vector $K$ in the attention mechanism. We introduce RoIPool to pool the RoI region. Third, we should calculate the attention score for matching the RoI region with other regions of interest. We also utilize the attention mechanism to assess the score. The fourth final step is the attention output, which aims to multiply the feature map obtained by the RoI region on the original image with the attention tensor to acquire the feature map $O$ of the region of interest. The specific algorithm is acquire in algorithm 1.

### C. Scene Text Classification

The obtained image features from the above steps are transformed into probabilities of context-aware scene text shape

**Algorithm 1** Context-aware RoI feature algorithm

**Input:** An image with a size of $H \times W \times C$, the target region of interest $RoI$, and the attention key-value pair $K$, $V$;
**Output:** Feature map of the region of interest $O6$

$X \leftarrow$ ResNet(image)
**function** QUERY_OUTPUT($X, RoI$)
  $x \leftarrow$ RoIPool$(X, RoI)$
  $Q \leftarrow$ Conv$(x.\text{shape}[1], d, \text{kernel_size} = 1)(V)$
  **return** $Q$
**end function**
$A \leftarrow$ Attention$(Q, K, V)$
**function** ATTENTION_OUTPUT($A, V$)
  $V1 \leftarrow$ Conv$(V.\text{shape}[1], C, \text{kernel_size} = 1)(V)$
  $V2 \leftarrow$ BN$(C)(V1)$
  $V3 \leftarrow$ ReLU$(V2)$
  **return** $V3$
**end function**
$output\_size \leftarrow (H, W))$
$V\_RoI \leftarrow$ RoIAlign($V3$,RoI, $output\_size$)
$O \leftarrow V\_RoI \times A$.unsqueeze$(1)$
$O1 \leftarrow$ Conv$(O, C)$
$O2 \leftarrow$ BN$(O1)$
$O3 \leftarrow$ ReLU$(O2)$
$O4 \leftarrow$ Conv$(O3, C)$
$O5 \leftarrow$ BN$(O4)$
$O6 \leftarrow$ ReLU$(O5)$
**return** $O6$

categories, enabling rapid classification of multi-category scene texts. The steps for multi-class context aware scene text classification are defined as follows:

$$f1 = flaten(O) \in \mathbb{R}^{HWC} \tag{6}$$

$$z1 = (f1W_1^T + b_1) \in \mathbb{R}^D \tag{7}$$

$$a1 = ReLU(z1) \in \mathbb{R}^D \tag{8}$$

$$d1 = Dropout(a1, p1) \in \mathbb{R}^D \tag{9}$$

$$y = (d1W_2^T + b_2) \in \mathbb{R}^{C'} \tag{10}$$

$$p_j = \frac{\exp(y_j)}{\sum_{k=1}^{C'} \exp(y_k)}, \quad j = 1, 2, \cdots, C' \tag{11}$$

Among them, the parameters of fully connected layer 1 and layer 2 are $W_1 \in \mathbb{R}^{HWC \times D}$, $b_1 \in \mathbb{R}^D$ and $W_2 \in \mathbb{R}^{D \times, C'}$, $b_2 \in \mathbb{R}^{C'}$, respectively, where $D$ represents the dimension of the hidden layer, $C'$ denotes the number of classes, $ReLU(0, x)$ defines the ReLU activation function, $Dropou(x, p)$ means the random inactivation operation, and $flatten(O)$ is flattening the three-dimensional tensor into a one-dimensional vector, where $p$ represents the probability of inactivation.

Finally, $p_j$ denotes the probability distribution that the input region of interest belongs to the $j$-th class. The category with the highest probability is the result of the final perceptual classification.

The fully connected layer are adopted to calculate the classification score, and to normalize the result through the $softmax$ function to obtain the classification prediction probability $P$. The specific algorithm is defined as shown in Algorithm 2, where the "@" represents the matrix multiplication operation of the fully connected layer, and $Dropout$ prevents the model from overfitting, $W1$, $W2$, $B1$, and $B2$ are all fully connected layer parameters.

**Algorithm 2** Context-aware scene text classification algorithm

**Input:** Feature Tensor $O$ for Region of Interest, Classifier Parameters $W$, Bias $B$
**Output:** Class prediction probability $P$

$F1 \leftarrow$ flarren(O)
$Z1 \leftarrow F1@W1 + B1$
$A1 \leftarrow$ ReLU$(Z1)$
$O3 \leftarrow$ Dropout$(A1, P1)$
$Y \leftarrow D1@W2 + B2$
$P \leftarrow$ softmax$(Y)$

### *D. Classification confidence threshold*

Confidence scores for each category are computed using the softmax function on each vector output by the classification model. when input an image $x$, the classification model outputs a vector $y$ representing the probability score for each class. The probability score for the $i$-th is defined as:

$$softmax_i(y) = e^{y_i} / \sum_{j=1}^{C} e^{y_j} \tag{12}$$

Among them, $C$ indicates how many categories there are, $y_i$ denotes the $i$-th element of the vector $y$, and $e$ is a constant. $\sum_{j=1}^{c} e^{y_j}$ is the exponential sum of the scores of all categories.

Because we only divides the scene text into two categories, it has been verified by experiments that when the scene text is classified, the confidence threshold $\sigma$ for judging irregular text should be set in the interval $[0.3, 0.6]$, the confidence threshold $\sigma$ for judging regular text should be in the interval $[0.4, 0.7]$, the image features are uploaded to the cloud for detection, therefore the framework performs better in both accuracy and efficiency.

## IV. Adaptive multi-category scene text detection

The high-precision arbitrary-shape scene text detection model can achieve higher recognition accuracy at the cost of lower recognition efficiency.

In scene text detection, Bezier curve are used to describe the bounding box of text. The Bezier curve only needs to determine the position of a few control points to accurately determine the position and shape of the text.A Bezier curve controls

the shape of the curve by $n$ control points $P_0, P_1, ..., P_{n-1}$. As shown in Fig. 4, let $AD/AB = BE/BC = DF/DE = t$, and the trajectory of point $F$ is the Bezier curve.

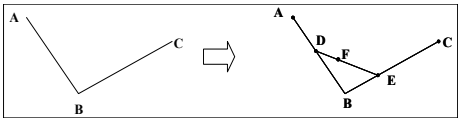


Fig. 4. Bezier curve trajectory plot.

The Bezier curve expression is as follows:

$$c(t) = \sum_{i}^{n} P_i B_{i,n}(t), 0 \leq t \leq 1 \tag{13}$$

$$B_{i,n}(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i}, i = 0, 1, 2..., n \tag{14}$$

where $n$ represents the number of control points, $P_i$ denotes the $i$-th control point, and $c(t)$ means the $n$-th Bezier curve.

Research shows that cubic Bezier curves are sufficient to fit arbitrarily shaped scene text, which is defined as:

$$c(t) = P_0(1-t)^3 + 3P_1(1-t)^2 + 3P_2t^2(1-t)^2 + P_3t^3, t \in [0, 1] \tag{15}$$

An adaptive multi-category scene text detection method is proposed based on the Bezier curve, which consists of several control points and curve segments. By changing the position of the control points, the shape and path of the curve can be precisely controlled. For the various shapes of scene text, this paper divides them into regular and irregular types. The detection strategy for irregular scene text is to utilize 8 control points to construct the shape of the Bezier curve.The construction process is presented in Fig. 5.

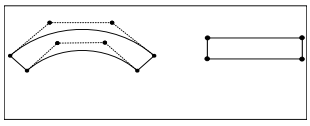

Fig. 5. Bézier control points generate curved renderings.

Therefore, a detection model for irregular text is trained on the cloud using the irregular scene text dataset and deployed to the edge. For regular scene text, due to its non-curved and Bezier curve characteristics, only 4 control points should be used to determine the shape of the Bezier curve during the training phase. Therefrom, through the above processes, the proposed method achieves performance improvement compared to existing methods in terms of model complexity and detection delay.

## V. EXPERIMENT

We evaluate our method on the arbitrarily shaped scene text benchmarks Total text [6] and CTW1500 [5] and the regular scene text benchmark ReCTS [7]. Meanwhile, by mixing a certain proportion of data from each of the three data sets, we constructed a data set with both regular scene texts and irregular scene texts.

### A. model recognition performance

Evaluation on the Total Text Dataset: The Total text [6] dataset is currently recognized authoritative arbitrarily shaped scene text benchmarks proposed in 2017, which is collected from a variety of scenes, including text-like scene complexity and low-contrast backgrounds. It includes 1555 images in which 1255 are for training and 300 for testing. Similar to real-world scenarios, most images in this dataset contain a large amount of regular text, while guaranteeing at least one curved text per image. Text instances are annotated with polygons based on word levels. The dataset contains only English text.

Table I presents the performance results of the proposed method on the Total text dataset. From Table I we observe that the method proposed is superior to the current advanced method in terms of Precision, Recall, F-measure and FPS. Some representative results are shown in Fig. 6.

TABLE I
COMPARISON OF DETECTION RESULTS ON THE TOTAL TEXT DATASET.

| **Method** | **Total text** | | | |
|---|---|---|---|---|
| | Precision | Recall | F-measure | FPS |
| TextBoxes [10] | 50.4 | 45.5 | 47.6 | 1.0 |
| TextDragon [11] | 75.3 | 68.9 | 73.6 | - |
| PSENet [12] | 82.3 | 78.6 | 81.3 | 3.4 |
| PAN [13] | 83.5 | 79.6 | 80.3 | 19.6 |
| ABCNet [4] | 81.8 | 78.4 | 81.4 | 14.5 |
| DBNet++ [15] | **85.6** | 80.1 | 82.3 | 19.6 |
| Ours | 84.5 | **81.2** | **84.2** | **20.3** |


Fig. 6. Display of recognition results on Total text.

Simulations on CTW1500 dataset: CTW1500 [5] contains 1500 high-quality text images, including 500 test images and 1000 training images. These images are from real-world scenes with various text forms, including Chinese and English characters, vertical, horizontal and curved text, etc.

Table II shows that our method achieves competitive performance compared with existing methods. Typical recognition examples for CTW1500 dataset are shown in Fig. 7.

Validations on the ReCTS dataset: ReCTS [7] is an open dataset for the Text Recognition task. The dataset contains more than 60,000 high-resolution text line images most of which come from different books, handwritten notes, and street view images. These lines of text include various languages

TABLE II
COMPARISON OF DETECTION RESULTS ON CTW1500 DATASETS.

| Method | CTW1500 | | | |
|---|---|---|---|---|
| | Precision | Recall | F-measure | FPS |
| TextBoxes [10] | 51.2 | 44.8 | 47.5 | 1.0 |
| CRAFT [16] | 85.3 | 77.8 | 82.2 | - |
| PSENet [12] | 82.1 | 78.2 | 80.3 | 3.3 |
| PAN [13] | 82.5 | 79.2 | 80.4 | 18.6 |
| ABCNet [4] | 82.6 | 80.2 | 81.8 | 15.2 |
| ContourNet [14] | 83.2 | 79.3 | 82.6 | 9.5 |
| DBNet++ [15] | **85.4** | 80.1 | **83.9** | 20.4 |
| Ours | 85.2 | **80.4** | 82.5 | **22.0** |

and text types, such as English, Chinese, Korean, Arabic. The ReCTS dataset is by far one of the largest datasets of text lines in multiple languages and fonts. The dataset also involves some common text recognition challenges, such as different fonts, poor lighting conditions, blurry images, and so on.

Table III compares the performance of the proposed method on the ReCTS dataset with the existing advanced scene text detection methods. It shows that our method achieves competitive performance in comparison with existing methods in terms of accuracy on the ReCTS dataset, since most of the scene text in ReCTS is regular scene text, the adaptive strategy of our model greatly shortens the time for the model to detect the scene text. Some recognition results are presented in Fig. 8.

TABLE III
COMPARISON OF DETECTION RESULTS ON RECTS DATASETS.

| Method | ReCTS | | | |
|---|---|---|---|---|
| | Precision | Recall | F-measure | FPS |
| PSENet [12] | 82.2 | 78.3 | 80.5 | 3.4 |
| PAN [13] | 83.1 | 79.8 | 80.6 | 19.6 |
| ABCNet [4] | 80.6 | 78.2 | 80.8 | 14.9 |
| ContourNet [14] | 83.9 | 80.3 | 82.6 | 9.2 |
| DBNet++ [15] | **87.4** | 82.1 | 84.2 | 20.3 |
| Ours | 87.2 | **82.4** | **85.1** | **28.0** |


Fig. 7. Display of detection results on CTW1500.

Verification on Mixed Dataset: To validate the performance of the proposed method on complex-shaped scene text datasets, we mixed the aforementioned three datasets in different proportions and shuffled the order. The performance of the proposed method is then evaluated on the mixed dataset


Fig. 8. Display of detection results on ReCTS.

as shown in TABLE IV. The results demonstrate that the proposed framework can significantly reduce the detection delay.

The mixed dataset consists of 3000 images from the three datasets, mixed in varying proportions for model performance testing.

Each of the three data sets,Total text, CTW1500, ReCTS, accounts for one-third of all image data.

TABLE IV
COMPARISON OF DETECTION RESULTS ON 1/3, 1/3, 1/3 MIXED-TYPE DATASETS.

| Method | 1/3,1/3,1/3 | | | |
|---|---|---|---|---|
| | Precision | Recall | F-measure | FPS |
| PSENet [12] | 81.8 | 78.5 | 80.6 | - |
| PAN [13] | 84.2 | 80.8 | 81.6 | 18.5 |
| ABCNet [4] | 83.6 | 81.3 | 81.8 | 14.9 |
| ContourNet [14] | 84.4 | 79.8 | 81.8 | 10.1 |
| DBNet++ [15] | 85.4 | 81.1 | 83.6 | 19.4 |
| Ours | **86.3** | **82.4** | **85.2** | **25.1** |

Both of the Total text, CTW1500 datasets account for a quarter of all image data and ReCTS dataset accounts for 1/2. The simulation results are shown in Table V.

TABLE V
COMPARISON OF DETECTION RESULTS ON 1/4, 1/4, 1/2 MIXED-TYPE DATASETS.

| Method | 1/4,1/4,1/2 | | | |
|---|---|---|---|---|
| | Precision | Recall | F-measure | FPS |
| PSENet [12] | 81.4 | 78.1 | 80.2 | - |
| PAN [13] | 84.0 | 81.1 | 81.5 | 19.1 |
| ABCNet [4] | 83.2 | 81.6 | 81.4 | 15.0 |
| ContourNet [14] | 84.1 | 80.0 | 80.8 | 10.2 |
| DBNet++ [15] | 85.1 | 80.9 | 83.7 | 20.0 |
| Ours | **86.4** | **82.1** | **85.2** | **26.4** |

Total text and CTW1500 dataset both account for 1/8, and ReCTS dataset accounts for 3/4. The validation results are shown in Table VI.

From the performance evaluation results of the above model on the mixed data set, it can be seen that the performance of our method achieve better performance compared with the existing advanced methods in terms of detection accuracy. For

TABLE VI
COMPARISON OF DETECTION RESULTS ON 1/4, 1/4, 1/2 MIXED-TYPE DATASETS.

| Method | 1/4,1/4,1/2 | | | |
|---|---|---|---|---|
| | Precision | Recall | F-measure | FPS |
| PSENet [12]] | 81.4 | 78.6 | 80.1 | - |
| PAN [13] | 84.4 | 81.0 | 80.9 | 19.0 |
| ABCNet [4] | 84.0 | 81.5 | 81.4 | 15.2 |
| ContourNet [14] | 84.0 | 79.1 | 81.4 | 9.5 |
| DBNet++ [15] | 84.8 | 81.3 | 83.4 | 20.0 |
| Ours | **86.8** | **83.0** | **85.5** | **27.1** |

detection delay, it is far superior. Moreover, with increasing the diversification degree of scene text shapes in the dataset, the performance of the proposed method presents an increasingly better trend in terms of accuracy and delay.

### *B. Adaptive detection verification*

The adaptive detection strategy proposed in this paper performs perceptual classification on the input image context in the model, and determines the shape category of the scene text. In the following, We verify the role of the proposed adaptive detection strategy in reducing model complexity. As above, the three datasets, CTW1500, ReCTS, and mixed dataset are all utilized. We mainly compare the delay consumed by the model in detecting scene text to evaluate the performance improvement brought by the adaptive detection strategy to the model.

Table VII shows the ablation experiments of the adaptive detection strategy proposed in this paper. The method proposed is to adaptively select the optimal detection strategy according to the different categories of scene text, so as to lower the scene text detection delay. The proposed model designs two detection strategies for adaptive selection, namely non-rule detection strategy and rule detection strategy. From the results, it can be observed that the adaptive detection effectively reduces the delay of recognition.

TABLE VII
COMPARISON OF DETECTION RESULTS OF ADAPTIVE DETECTION STRATEGY.

| Method | CTW1500 | | | | ReCTS | | | |
|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | F | FPS | Precision | Recall | F | FPS |
| Non-rule | 85.1 | **80.1** | **84.1** | 18.5 | **86.4** | **81.3** | 85.2 | 17.8 |
| Rule | 82.1 | 79.2 | 80.6 | **24.6** | 85.3 | 81.4 | 82.6 | **30.2** |
| Adaptive | **85.2** | 80.4 | 84.1 | 22.0 | **87.2** | 82.4 | 85.1 | 28.0 |

## VI. CONCLUSION

In this paper, we propose a context-aware edge-cloud collaborative scene text recognition method. In multi-category mixed scene text recognition, by context-aware classification of input images or videos, the optimal detection strategy is adaptively adopted according to the category of scene text, thereby reducing model complexity and computational overhead. In order to balance the recognition accuracy and recognition delay, we introduce an edge-cloud collaborative architecture, which integrates the computing resource of the cloud layer and edge layer for collaborative scene text detection. Finally, the simulation experiments show that the architecture and algorithm designed can significantly improve the recognition delay while ensuring the recognition accuracy.In the future, we plan to combine the deep reinforcement learning model to design an adaptive framework that is tightly coupled with the actual recognition scene to further reduce the delay.

## REFERENCES


[1] W. Feng, F. Yin, X. -Y. Zhang and C. -L. Liu, "Semantic-Aware Video Text Detection," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 1695-1705.

[2] J. Tang et al., "Few Could Be Better Than All: Feature Sampling and Grouping for Scene Text Detection," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 4553-4562, doi: 10.1109/CVPR52688.2022.00452.

[3] Qiao, Liang, et al. "Text perceptron: Towards end-to-end arbitrary-shaped text spotting." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 07. 2020.

[4] Y. Liu, H. Chen, C. Shen, T. He, L. Jin and L. Wang, "ABCNet: Real-Time Scene Text Spotting With Adaptive Bezier-Curve Network," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 9806-9815.

[5] Liu Y, Jin L, Zhang S, et al. Curved scene text detection via transverse and longitudinal sequence connection[J]. Pattern Recognition, 2019, 90: 337-345.

[6] C. K. Ch'ng and C. S. Chan, "Total-Text: A Comprehensive Dataset for Scene Text Detection and Recognition," 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 2017, pp. 935-942, doi: 10.1109/ICDAR.2017.157.

[7] Zhang R, Zhou Y, Jiang Q, et al. Icdar 2019 robust reading challenge on reading chinese text on signboard[C]//2019 international conference on document analysis and recognition (ICDAR). IEEE, 2019: 1577-1581.

[8] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[9] Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 10012-10022.

[10] Liao M, Shi B, Bai X, et al. Textboxes: A fast text detector with a single deep neural network[C]//Proceedings of the AAAI conference on artificial intelligence. 2017, 31(1).

[11] Feng W, He W, Yin F, et al. Textdragon: An end-to-end framework for arbitrary shaped text spotting[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 9076-9085.

[12] W. Wang et al., "Shape Robust Text Detection With Progressive Scale Expansion Network," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 9328-9337, doi: 10.1109/CVPR.2019.00956.

[13] W. Wang et al., "Efficient and Accurate Arbitrary-Shaped Text Detection With Pixel Aggregation Network," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 8439-8448, doi: 10.1109/ICCV.2019.00853.

[14] Y. Wang, H. Xie, Z. -J. Zha, M. Xing, Z. Fu and Y. Zhang, "ContourNet: Taking a Further Step Toward Accurate Arbitrary-Shaped Scene Text Detection," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 11750-11759, doi: 10.1109/CVPR42600.2020.01177.

[15] M. Liao, Z. Zou, Z. Wan, C. Yao and X. Bai, "Real-Time Scene Text Detection With Differentiable Binarization and Adaptive Scale Fusion," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 1, pp. 919-931, 1 Jan. 2023, doi: 10.1109/TPAMI.2022.3155612.

[16] Y. Baek, B. Lee, D. Han, S. Yun and H. Lee, "Character Region Awareness for Text Detection," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 9357-9366, doi: 10.1109/CVPR.2019.00959.