# CAMTNet: CTC-Attention Mechanism and Transformer Fusion Network for Scene Text Recognition

Ling Wang, *Member, IAENG*, Kexin Luo, *Member, IAENG*, Peng Wang, and Yane Bai

***Abstract*—Current scene text recognition models excel in recognizing regular text images, yet there remains a need for advancements in identifying irregular text images. In this paper, we propose the challenge by introducing CAMTNet, a novel text recognition model based on Convolutional Recurrent Neural Network (CRNN). CAMTNet includes a rectification module for irregular text images. In addition, VGGNet was replaced by ResNet, which has a fused Coordinate Attention mechanism to improve feature comprehension. Furthermore, the model utilizes the Transformer as the encoder module, in order to capture contextual information for improved feature extraction. In the decoder module, we combine the Connectionist Temporal Classification with the sequence-based attention mechanism, to improve the model's contextual information capturing and sequence decoding capabilities. CAMTNet outperforms CRNN across six benchmark scene text recognition datasets, achieving a 7% increase in average recognition accuracy on three regular datasets, and a notable 20% increase on three irregular datasets.**

***Index Terms*—Scene text recognition, CRNN, Feature extraction, Transformer, Coordinate Attention mechanism**

## I. Introduction

Scene text recognition (STR) plays a crucial role in the domain of scene text analysis. In recent years, machine learning and deep learning technologies are maturing [1]. In many deep learning systems, reading text in natural scene images contributes to the development of various fields, including robot navigation, industrial automation, and driver assistance [2]. Despite the increasing scholarly attention towards text recognition in diverse scenes, the field of STR encounters persistent challenges.

Some traditional methods first locate individual characters, and then utilize Convolutional Neural Networks (CNN) to recognize each cropped character, completing the recognition of text in the scene images. While such methods are beneficial for recognizing individual character images, they fail to capture the correlations between characters. With the rapid development of deep learning, STR models [3, 4] such as CRNN are proposed. CRNN can handle sequences of arbitrary lengths, and demonstrate significant recognition competence on both dictionary-based and non-dictionary-based datasets. However, despite the advantages of CRNN, it faces two major issues. Firstly, CRNN exhibits suboptimal performance in irregular text images. Secondly, CRNN is less effective than complex models in recognizing irregular text images, such as curved, distorted, and low-quality ones.

In view of the above issues, the main contributions of this paper are threefold:

(1) To improve the recognition accuracy of irregular text images, this paper combines the combination of Structure-Preserving Inner Offset Network (SPIN) [5] with Thin-Plate Spline (TPS), in order to correct curved and uneven color text, and partially restore distorted images.

(2) To improve the recognition accuracy of STR models, we make several key improvements to the CRNN architecture. Initially, we swapped out the standard feature extraction network for a more robust ResNet, complemented by the Coordinate Attention (CA) mechanism to spotlight pivotal features. Additionally, we incorporate the Transformer [6] to capture broader semantic insights, which are then seamlessly woven into the feature sequence, significantly boosting recognition capabilities. Lastly, we fuse the Connectionist Temporal Classification (CTC) with a Sequence-to-Sequence attention mechanism, refining the decoding process by factoring in contextual and local details. This comprehensive approach ensures precise text recognition, even against complex image backgrounds.

(3) Experiments conducted on benchmark test datasets demonstrate that CAMTNet outperforms existing models regarding recognition accuracy. Notably, it exhibits strong robustness in recognizing low-quality and irregular text images in STR.

## II. Related Work

While numerous models are proposed for STR, several face challenges in achieving the requisite accuracy for recognizing irregular text in industrial environments. This can be attributed to either inadequate performance or the inherent complexity of their structures, rendering them unsuitable for real-world applications.

In response to the challenge of accurately recognizing irregular text images, numerous researchers suggest

Manuscript received January 18, 2024; revised September 14, 2024.

Ling Wang is a lecturer at the School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022 China. (corresponding author phone: 138-0431-0918; e-mail: wangling0912@cust.edu.cn).

Kexin Luo is a graduate student of Changchun University of Science and Technology, Changchun 130022 China. (e-mail: 992098930@qq.com).

Peng Wang is a professor at the School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022 China (e-mail: wangpeng@cust.edu.cn).

Yane Bai is a lecturer at the School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022 China. (e-mail: bye1023@cust.edu.cn).

incorporating a rectification module into models to address this issue. Luo et al. [7] propose a Multi-Object Rectified Attention Network (MORAN), which employs a pixel-level weakly supervised learning mechanism specifically tailored for rectifying irregular texts. However, MORAN exhibits limitations in effectively recognizing highly curved text images and poses challenges in training. Shi et al. [8] propose adding Spatial Transformer Networks (STN) [9] before the feature extraction network to correct irregular images, resulting in improved recognition accuracy for curved texts. Subsequently, several models, such as those presented in [10-12], adopt STN as a method for text rectification to improve recognition accuracy. Shi et al. [13] suggest utilizing a module that combines STN and TPS as the text rectification module. TPS offers precise geometric transformations and maintains a simple structure, thereby minimizing the impact on the model's computational speed.

The above methods primarily focus on rectifying irregular images, but there remains a need to enhance the rectification efficacy for low-quality text images with uneven colors. To solve the problem, Zhang et al. introduce a neural network structure based on SPIN, this approach divides the input image into multiple subregions for feature extraction and text positioning tasks, followed by interpolating and fusing the results. Therefore, to address the recognition problem of low-quality and irregular text images, this paper incorporates a module that combines SPIN and TPS to correct the text. This integration enables the model to rectify irregular texts and improve color distortion of text images within scenes.

In order to improve the text recognition performance, several structurally complex models are proposed. Lu et al. [14] present the Multi-Aspect Scene Text Recognition (MASTER) model, which utilizes a global contextual attention mechanism encoder and a Transformer-based decoder. While MASTER boosts recognition speed through parallel computing, its accuracy for low-resolution text images is subpar, and it struggles with text recognition in complex scenes. Du et al. [15] propose an end-to-end scene text recognition model that employs a unified visual model for both feature extraction and text transcription, leading to improved accuracy and speed. However, this model's reliance solely on visual cues overlooks semantic text information to some extent, hindering effective contextual association. Hu et al. [16] combine the attention mechanism and the CTC to guide features during training to avoid inference time slowdown. Nevertheless, due to the inference time slowdown caused by the attention mechanism, the attention branch is employed only during the training phase. In addition, the model requires preprocessing before use, which incurs a high time cost and makes it difficult to apply in actual scenarios. Therefore, this paper leverages a Transformer encoder and a CTC-Attention decoder to capture and analyze contextual information in text images. This approach aims to augment the model's ability to extract semantic information without compromising its structural simplicity.

## III. Methodology

As shown in Figure 1, CAMTNet consists of four main parts. Initially, the rectification module, comprising SPIN and TPS, corrects distortions in irregular text images. Following this, the feature extraction network, which includes a residual convolutional network and the CA, is tasked with extracting visual features from the text images. The encoder module, the third part, employs a multi-head attention mechanism and a Feedforward Neural Network to capture contextual information from the extracted features. Lastly, the decoder module integrates CTC with a Sequence-to-Sequence attention mechanism to predict the character probability distribution. The details of each module are presented in subsequent sections.

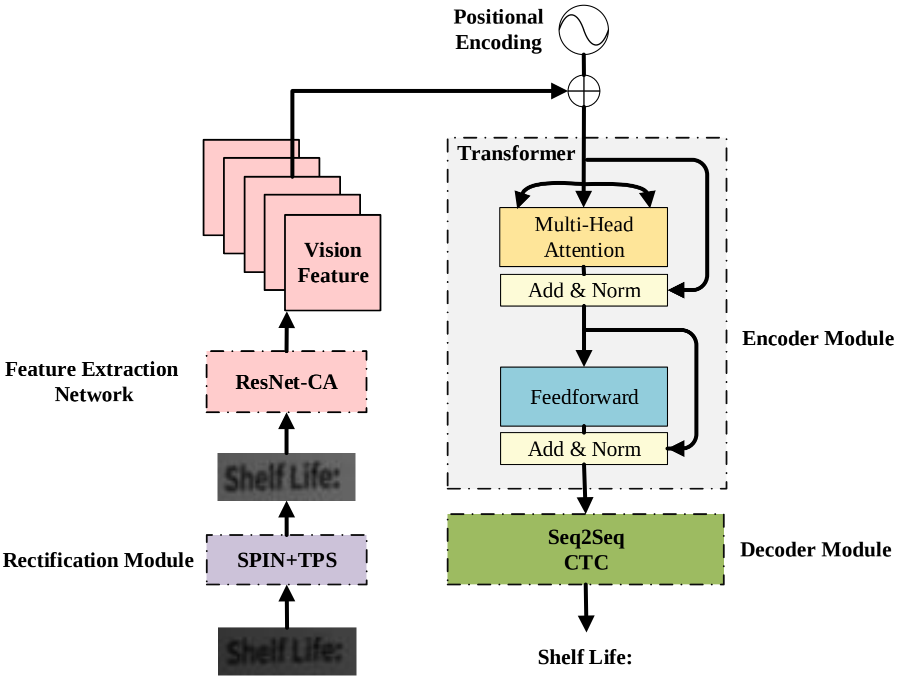


Fig. 1. Overview of CAMTNet. CAMTNet contains a rectification module, a feature extraction network, an encoder module and a decoder module.

### A. *Rectification Module*

This paper introduces a rectification module to precede the feature extraction network, aiming to enhance the recognition accuracy of irregular text images, particularly those affected by uneven lighting, curved characters, and poor quality. The rectification module combines SPIN and TPS, which can improve the correction of irregular text images with uneven coloration can be achieved.

As demonstrated in Figure 2, the rectification module initially employs SPIN for color correction, it subsequently utilizes TPS to straighten the curved text into a horizontal position.

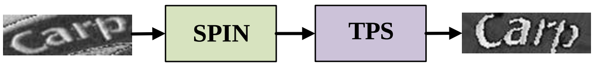


Fig. 2. The structure of Rectification Module.

The structure of TPS is shown in Figure 3.

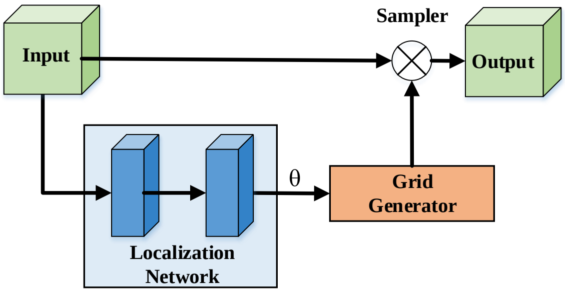


Fig. 3. The structure of TPS.

TPS consists of a Localization Network and a Grid Generator. The Localization Network calculates the transformation parameters essential for the input text image. Subsequently, the Grid Generator generates a coordinate grid, using the transformed parameters to resample pixels and rectify text irregularities in the image.

The structure of SPIN is depicted in Figure 4. SPIN contains Structure Preserving Network (SPN) and Auxiliary Inner-offset Network (AIN). The SPN deftly handles the common challenge of inconsistent color distribution in text images. The AIN efficiently identifies variations in color intensity between individual characters. Initially, SPIN thoughtfully partitions the image into small blocks, and then predicts the offset for each block. These calculated offsets are then skillfully activated using the Sigmoid function, followed by scaling up to match the dimensions of the input image through an upsampling operation.

### B. *Feature Extraction Network*

VGGNet serves as the feature extraction network in CRNN. Despite its structure being simple and easy to train, its capability to extract features from intricate text images is considered inadequate. To address this limitation, this paper replaces VGGNet with ResNet, leveraging its more sophisticated convolutional architecture to extract more informative features from complex text images. Simultaneously, to improve the model's feature extraction capability, the paper introduces the CA mechanism in ResNet. Detailed configurations of the feature extraction network are presented in TABLE I.

TABLE I
FEATURE EXTRACTION NETWORK CONFIGURATIONS OF CAMTNET

| Layers | Configurations | Outputs |
|---|---|---|
| Conv1<br>Black1 | 3×3conv,stride1×1,32<br>$\begin{bmatrix}3\times3,32\\3\times3,32\\\text{CA-mechanism}\end{bmatrix}\times3, stride2\times2$ | 32×32×100<br>32×16×50 |
| Conv2<br>Black2 | 3×3conv,stride1×1,32<br>$\begin{bmatrix}3\times3,64\\3\times3,64\\\text{CA-mechanism}\end{bmatrix}\times4, stride2\times2$ | 32×16×50<br>64×8×25 |
| Conv3<br>Black3 | 3×3conv,stride1×1,64<br>$\begin{bmatrix}3\times3,128\\3\times3,128\\\text{CA-mechanism}\end{bmatrix}\times6, stride2\times1$ | 64×8×25<br>128×4×25 |
| Conv4<br>Black4 | 3×3conv,stride1×1,128<br>$\begin{bmatrix}3\times3,256\\3\times3,256\\\text{CA-mechanism}\end{bmatrix}\times6, stride2\times1$ | 128×4×25<br>256×2×25 |
| Conv5<br>Black5 | 3×3conv,stride1×1,256<br>$\begin{bmatrix}3\times3,512\\3\times3,512\\\text{CA-mechanism}\end{bmatrix}\times3, stride2\times1$ | 256×2×25<br>512×1×25 |

The feature extraction network comprises five 3×3 convolution modules (Conv) with a 1×1 stride and five stacked residual attention blocks (Block). The first two blocks of Block have a 2×2 stride, while the following three blocks use a 2×1 stride. The model rescales the input image uniformly to 32×100, maintaining consistent input and output dimensions for Conv. Block configuration is shown in Figure 5.

As demonstrated in Figure 5, the Block consists of two convolution modules, a ReLU activation function, and a CA block. The convolution module includes a 3×3 convolution (Conv2d) and a batch normalization layer (BN). The CA block incorporates spatial information in both width and height dimensions by leveraging channel information, thus augmenting the feature extraction network's capacity for expression. The specific formulas for the average pooling results of features in both directions, denoted as $z^h$ and $z^w$, are given by Equations (1) and (2), respectively.

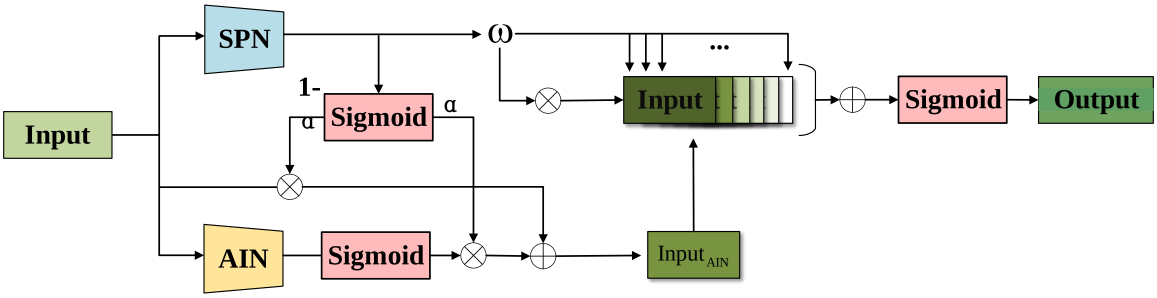


Fig. 4. The structure of SPIN.

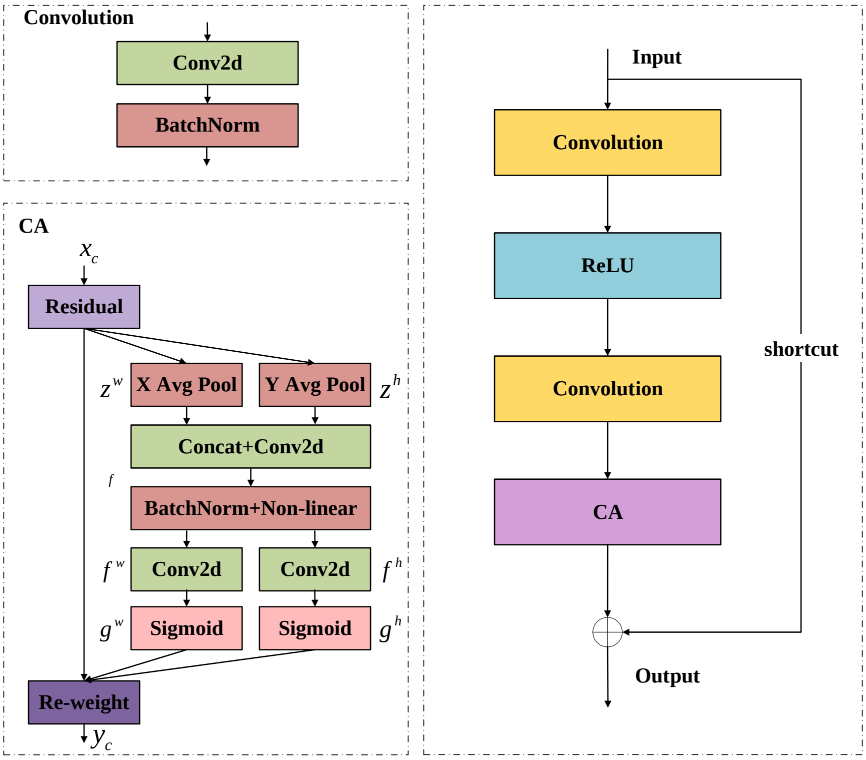


Fig. 5. The structure of Block. It consists of three parts, namely the convolution blocks, the ReLU and the CA blocks.

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

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

Where $H$ and $W$ represent the height and width of the input feature map, $x_c(h,i)$ is the vertical feature vector with a height of $h$, and $x_c(j,w)$ is the horizontal feature vector with a width of $w$.

Following the average pooling of $z^h$ and $z^w$, the feature map $f$ is derived by concatenation and convolution operations, as specified by Equation (3).

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

Where $F_1$ denotes the utilization of 1×1 convolution for dimensionality reduction and activation operations.

Next, the feature map $f$ is further processed to obtain two directional feature maps, denoted as $f^h$ and $f^w$. $f^h$ and $f^w$ undergo convolution operation $F$ for dimensionality expansion, followed by the application of the Sigmoid function to generate attention vectors $g_c^h$ and $g_c^w$ for distinct directions. Equations (4) and (5) define the formulas for $g_c^h$ and $g_c^w$ respectively.

$$g^h=\alpha(F_h(f^h)), \tag{4}$$

$$g^w=\alpha(F_w(f^w)). \tag{5}$$

Where $f^h\in R^{C/r\times H\times 1}$ and $f^w\in R^{C/r\times 1\times W}$ are vertical and horizontal feature maps respectively.

Ultimately, the attention vectors are fused with the output feature vector to yield the final output result, as depicted by Equation (6).

$$y_c(i,j)=x_c(i,j)\times g_c^h(i)\times g_c^w(j). \tag{6}$$

Where $x_c$ is the input feature vector, $y_c$ is the output feature vector from the CA block.

### C. Encoder Module

Bidirectional Long Short-Term Memory (BiLSTM) serves as the encoder module in CRNN. Nevertheless, this method may encounter challenges related to contextual awareness, hindering the effective capture of long-term and global dependencies among characters. Therefore, our proposed method uses the Transformer to master the long-term and global dependency relationships of sequence information, enhancing character recognition in text. The structure of the encoder module is shown in Figure 6.

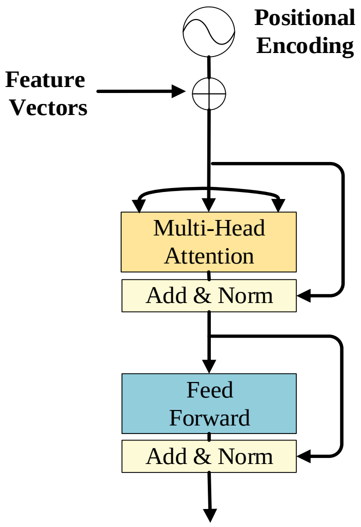


Fig. 6. The structure of Transformer.

Transformer consists of a Multi-Head Attention (MHA) and a Feedforward neural network. The specific architecture of MHA is illustrated in Figure 7.

MHA is a pivotal component, consisting of multiple self-attention mechanisms. In each self-attention mechanism, a query vector interacts with a collection of key-value pairs. Through the computation of attention scores between each

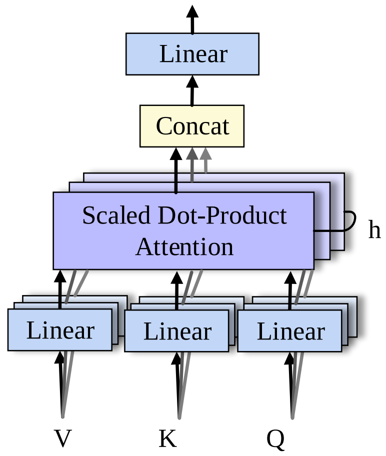


Fig. 7. The structure of Multi-Head Attention (MHA).

query vector and the keys, a weighted vector is generated as an output. Each attention mechanism in the MHA possesses distinct weight parameters, facilitating the capture of various relationship aspects. This structure allows the model to simultaneously attend to various segments of the input sequence, capturing dependencies across different positions.

In Transformer, the Feedforward neural network is a fully connected network with two linear layers. It helps the module extract higher-level feature representations through linear transformations and non-linear activation functions. This process enhances the model's performance to convert contextual information into more efficient feature representations, thereby facilitating subsequent task processing. The specific formula is presented in Equation (7).

$$FFN(x)=\max(0,xW_1+b_1)W_2+b_2 \qquad (7)$$

Where $x$ represents the input, while $W_1$, $W_2$, $b_1$, and $b_2$ all denote learnable parameters.

To ensure feature consistency across different layers, the Transformer incorporates normalization layers following both the MHA and the Feedforward neural network. Two frequently used normalization layers in Transformer are Layer Normalization (LN) and Batch Normalization (BN). Experimental findings indicate that LN often performs better than BN in STR. Consequently, this study chose LN as the normalization layer in CAMTNet.

In addition, to effectively address gradient propagation challenges and accelerate the model's convergence rate, residual connection layers are incorporated into the Transformer. These layers facilitate a seamless information flow between modules, mitigating issues related to gradient fading and exploding during training. Furthermore, they also ensure effective information transfer between successive layers, thus enhancing the overall performance of the network.

### *D. Decoder Module*

In STR, the decoder module plays a pivotal role in sequence analysis and decoding, typically falling into two categories: CTC and sequence-based attention mechanism. CRNN employs CTC as a decoder module. However, CTC assumes that feature sequences are independent of each other, which limits the ability of a single CTC to fully exploit context information for decoding analysis.

The sequence-based attention mechanism can effectively suppress irrelevant information and enhance the emphasis on pertinent details through contextual analysis. However, the single sequence-based attention mechanism may not guarantee alignment of the output. Alignment of the output to match the character length in the input text. Hence, the decoder module of CAMTNet combines both the CTC and the sequence-based attention mechanism. The structure is illustrated in Figure 8.

Figure 8 indicates that, the contextual sequence information of length $h=(h_1,h_2,...h_t)$ output by the encoder module is input to both the sequence-based attention mechanism and the CTC, leading to the generation of the predicted sequence $c=(c_1,c_2,...c_t)$ . The CTC provides a

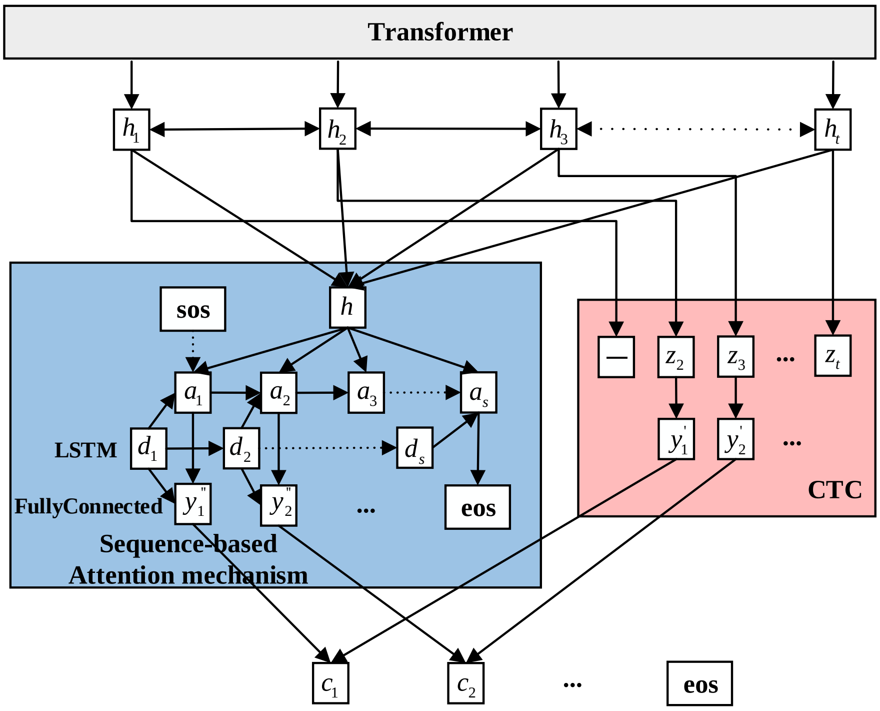


Fig. 8. The structure of the Decoder Module.

technique to train sequences without requiring pre-alignment of data. The probability $p_C(y^{'}|h)$ in this part is computed as shown in Equation (8).

$$p_C(y^{'}|h)=\sum_{z\in\theta(y^{'})}p_C(z|h). \tag{8}$$

Where $\theta(y^{'})$ represents the set of all possible paths that could map to $y^{'}$ in the context of the CTC. The calculation formula for $p_C(z|h)$ is given by Equation (9).

$$p_C(z|h)=\prod_{t=1}^{t}q_t^{z_t}. \tag{9}$$

Where $z=(z_1,z_2,...z_t)$ denotes the path for the CTC, and $q_t^{z_t}$ represents the maximum output path for the output label at time $t$.

From the above formulas, it can be observed that the probabilities calculated by the CTC are based on the assumption of independence between characters, which neglects global information and fails to better predict long textual sequences.

On the other hand, the sequence-based attention mechanism directly predicts the probability of the text sequence, utilizing the previous output results $y^{''}_{1:s-1}$ and the input of features $h$ from the previous module to obtain the output sequence $y=(y_1^{''},y_2^{''},...,y_s^{''})$. The detailed calculation for the sequence-based attention mechanism in Figure 8 is depicted in Equation (10).

$$\begin{aligned}
&p_A(y^{''}|h)=\prod_{s}p(y^{''}_{s}|h,y^{''}_{1:s-1}),\\
&y^{''}_{s}=FullyConnected(d_s,a_s),\\
&d_s=LSTM(d_{s-1},y^{''}_{s-1},a_s),\\
&a_s=\sum_{t}\in_{s,t}h_t.
\end{aligned} \tag{10}$$

Where $p(y_s^{''}|h,y^{''}_{1:s-1})$ denotes the prediction probability at the current time step, obtained from input features $h$ and the preceding $y^{''}_{1:s-1}$ output labels. $d_s$ denotes the hidden state of the LSTM, and $a_s$ is the context vector based on input features and attention weights.

Finally, the combination of $p_C(y^{'}|h)$ from the CTC and $p_A(y^{''}|h)$ from the sequence-based attention mechanism results in the prediction at the current time step $t$. The specific calculation is detailed in Equation (11).

$$c=\mathop{\arg\max}_{y^{'}\in \mathrm{D}}\{\alpha\log p_C(y^{'}|h)\}+(1-\alpha)p_A(y^{''}|h). \tag{11}$$

Where $D$ represents the character dictionary, and $\alpha$ in the formula is a variable parameter used to balance the weights between the CTC and the sequence-based attention mechanism. Its values range from 0 to 1. When $\alpha=1$, it indicates decoding using only CTC, and when $\alpha=0$, it indicates transcription using only the sequence-based attention mechanism.

### E. Loss Function

The loss calculation includes the CTC loss $L_C$ and the loss from the sequence-based attention mechanism $L_A$, as illustrated in Equation (12).

$$L=\alpha L_C+(1-\alpha)L_A. \tag{12}$$

Experimental results show that CAMTNet performs optimally when the value of $\alpha$ is set to 0.6. Further elaboration of the experimental results will be provided in the ablation experiment.

The CTC loss $L_C$ is calculated in Equation (13).

$$L_C=-\sum_{(x,y)\in S}p(y|x). \tag{13}$$

where $S$ denotes the training set, $\sum_{(x,y)\in S}$ denotes the sum of each sample probabilities of the output sequence given the input sequence, $p(y|x)$ represents the probability that the input sequence $x$ is the temporal output sequence $y$. The attention mechanism loss $L_A$ is calculated based on the Cross Entropy loss function, it is shown in Equation (14).

$$L_A(\hat{x},x)=-\sum_{i=1}^{n}x\log(\hat{x}). \tag{14}$$

where $\hat{x}$ represents the model's prediction corresponding to the ground truth $x$.

## IV. Experimental Results

In this section, we introduce the text image datasets, environment configurations, and model evaluation metrics. Subsequently, we evaluate CAMTNet's performance through comparative experiments.

### A. Datasets

CAMTNet is optimized on MJSynth [17] and SynthText [18]. To demonstrate the effectiveness of our model, we use six scene text recognition datasets. This section provides a brief introduction in the following paragraphs.

MJSynth (MJ) [17]: a synthetic dataset containing 8.9 million images of natural scenes in complex scenes. It is widely used for training recognition models.

SynthText (ST) [18]: a synthetic dataset contains approximately 5.5 million synthetic images. These images are used to extract text to help models train.

IIIT5K [19]: includes a training subset of 2000 samples and a validation subset of 3000 samples.

ICDAR 2013 (IC13) [20]: contains 1015 scene text images, most of which are regular text images cropped from the mall.

Street View Text (SVT) [21]: contains 647 low-quality word images taken from Google Street View.

Street View Text Perspective (SVTP) [22]: includes 645 test images cropped from SVT. Most of the images are corrupted by blur and low resolution.

CUTE80 (CT) [23]: contains 288 word images, most of which are irregular text images.

ICDAR 2015 (IC15) [24]: contains 2077 text images captured incidentally by Google Glasses.

Example images from the datasets are illustrated in Figure 9.

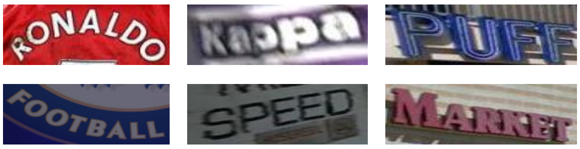


Fig. 9. Example plots of the datasets.

### B. Implement Details

The experimental software and hardware environment in this paper are detailed in TABLE II.

TABLE II
EXPERIMENTAL ENVIROMENT

| Name | Model/vision |
|---|---|
| CPU | Intel(R) i7-10700K CPU |
| GPU | NVIDIA GeForce RTX 2080Ti |
| Network framework | Pytorch 1.12.1 |
| Python | 3.9 |

The settings of parameters during model training are presented in TABLE III.

TABLE III
THE TRAINING PARAMETERS OF THE MODEL

| Name | Model/vision |
|---|---|
| batch_size | 256 |
| lr | 1 |
| epoch | 100 |
| optimizer | Adadelta[25] |

### C. Evaluation Index

This paper employs recognition accuracy and FPS as the evaluation metric. Recognition accuracy is defined as the ratio of correctly recognized text images to the total number of text images, which is calculated as shown in Equation (15).

$$\text{Accuracy} = \frac{N_r}{N_{all}}. \tag{15}$$

Where $N_r$ represents the number of text images correctly recognized by the model, and $N_{all}$ represents the total number of text images recognized by the model. Higher recognition accuracy indicates better performance of the model.

FPS refers to the frame rate when models recognize the text image, which is calculated as shown in Equation (16).

$$\text{FPS}=\frac{N}{\sum_{0<i<N}(t_{iend}-t_{istart})} \tag{16}$$

Where $i$ represents the ith image in the test dataset, $N$ represents the test dataset containing text images, $t_{istart}$ denotes the time of the ith image starts to be recognized, and $t_{iend}$ denotes the time of the ith image ends recognized. Larger FPS represents the faster recognition speed of the model.

### D. Comparsion Experiment

In this section, CAMTNet is compared with other advanced models to verify recognition competence. Comparative results are shown in TABLE IV, the bold text indicates the best result, and the underlined text indicates the second best result.

The comprehensive evaluation reveals that CAMTNet demonstrates a remarkable improvement in text recognition accuracy when compared to CRNN. The average recognition accuracy of CAMTNet stands at an impressive 94.9% on regular datasets and a commendable 86.5% on irregular datasets. Specifically, CAMTNet achieves an increase in recognition accuracy of 6.7%, 5.1%, and 9.2% on regular datasets than CRNN. More impressively, on irregular datasets, which are often more challenging due to their varied and complex nature, CAMTNet's accuracy enhancements are even more pronounced, with improvements of 16.2%, 23.1%, and 20.7% than CRNN. In addition, compared with SVTR-T, regarding the recognition performance on irregular datasets, although the recognition accuracy of CAMTNet exhibits a slightly lower recognition accuracy on the IC13, it surpasses SVTR-T by 3.1% and 0.4% on IIIT5K and SVT, respectively. The experimental results further confirm the efficacy of CAMTNet, showcasing improved performance in scene text recognition tasks through the collaborative synergy of its various modules.

Meanwhile, the FPS of CAMTNet reaches 12.83, which is 2.61 lower than that of CRNN and 1.18 higher than that of SVTR-T. Compared to other models, the inference speed of this model also remains at the upper middle level. Considering that the recognition speed is also an important reference standard in the scene text recognition task, although the model is not optimal in the inference speed, its FPS value meets the requirements of scene text recognition, and its better recognition accuracy makes the model better applied to the scene text recognition.

Additionally, to better showcase the superiority of CAMTNet in recognizing irregularities within the datasets, this section selectively extracts some irregular text images from the test datasets for visual comparison.

The recognition results between SVTR-T and CAMTNet are illustrated specifically in Figure 10.

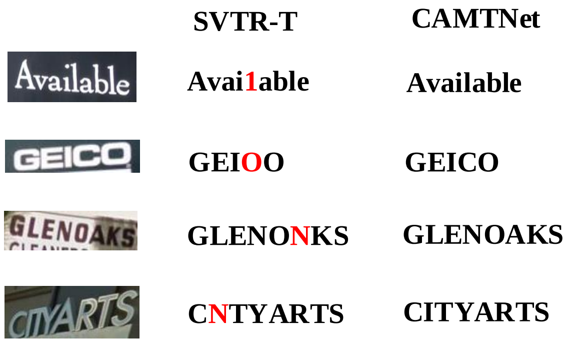


Fig. 10. Visualizing the results between SVTR-T and CAMTNet.

The detailed analysis presented in Figure 10 provides a clear visual comparison between the performance of CAMTNet and SVTR-T on a set of example images. It is evident that CAMTNet successfully recognizes all the text without errors, demonstrating its robustness and accuracy in text recognition tasks. In contrast, the SVTR-T model exhibits recognition errors in these example images. Notably, the model misidentifies 'l' as '1' in the first example, 'C' as 'O' in the second, 'A' as 'N' in the third, and 'I' as 'B' in the last. These errors suggest that SVTR-T may not be as effective in handling the nuances of text recognition, particularly when the text is presented in less-than-ideal conditions or formats.

TABLE IV
COMPARSION OF RECOGNITION ACCURACY WITH SOTA MODELS.

| Model | Accuracy(%) | | | | | | FPS |
|---|---|---|---|---|---|---|---|
| | Regular | | | Irregular | | | |
| | IC13 | IIIT5K | SVT | SVTP | CT | IC15 | |
| CRNN[3] | 89.1 | 91.8 | 82.7 | 69.9 | 65.4 | 64.1 | **15.44** |
| SRN[26] | 94.7 | 93.4 | 90.6 | 85.3 | 86.6 | 81.5 | 10.51 |
| VisionLAN[27] | 94.8 | 94.9 | 91.2 | 85.9 | 87.5 | 83.4 | 13.91 |
| MASTER[14] | 95.1 | 94.7 | 90.3 | 83.8 | 86.9 | 79.2 | 12.86 |
| SVTR-T[15] | **96.2** | 93.8 | 91.5 | 84.7 | 88.4 | 84.5 | 11.65 |
| VSDF[28] | 91.8 | 92.1 | 85.5 | 79.6 | 83.4 | 76.4 | 11.92 |
| NRSTRNet[29] | 92.8 | 88.5 | 88.6 | 82.0 | 76.0 | 73.1 | 14.37 |
| CAMTNet | 95.8 | **96.9** | **91.9** | **86.1** | **88.5** | 84.8 | 12.83 |

The recognition results between CRNN and CAMTNet are shown in Figure 11.

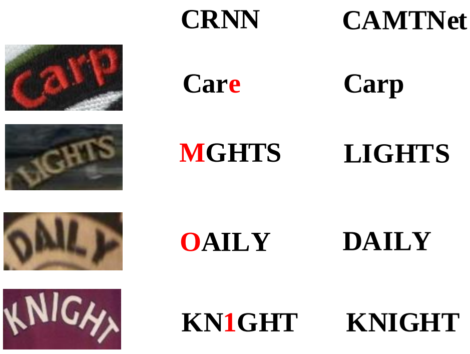


Fig. 11. Visualizing the results between CRNN and CAMTNet.

As demonstrated in Figure 11, the comparative analysis between CAMTNet and CRNN reveals significant differences in their performance, particularly when dealing with irregular text characters. CAMTNet consistently delivers accurate recognition across the board, confirming its robustness and reliability in text recognition tasks. Conversely, CRNN encounters challenges with irregular characters, as evidenced by the misidentification of 'p' as 'e', 'L' as 'M', 'D' as 'O', and 'I' as '1'. These errors suggest that CRNN's feature extraction and sequence analysis mechanisms may not be as refined as those of CAMTNet.

Upon thorough analysis and visualization of the experimental outcomes, it is evident that the text recognition capabilities of the CAMTNet significantly outperform those of the SVTR-T and CRNN. The model's ability to accurately process text images marred by distortions, such as those caused by perspective skew, low resolution, and uneven lighting, demonstrates its robustness and potential for practical applications in various real-world settings.

## V. Ablation Study

In order to better verify the validity of CAMTNet, the performance of each module is evaluated separately in this section.

### A. The effectiveness of the Rectification Module

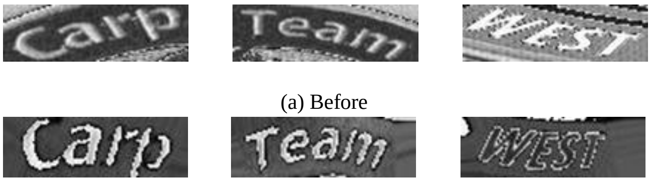
In this section, our work is to test the performance of CAMTNet before and after integrating the rectification module on regular and irregular datasets. The experimental results are shown in TABLE V.

TABLE V
The performance before and after adding rectification module on irregular datasets

| Model | Accuracy(%) | | |
|---|---|---|---|
| | SVTP | CT | IC15 |
| w/o rectification | 84.8 | 86.3 | 82.4 |
| only SPIN | 85.6 | 87.9 | 84.1 |
| only TPS | 75.3 | 87.4 | 83.7 |
| rectification | **86.1** | **88.5** | **84.8** |

From TALBE V, it can be observed that with the addition of the rectification module, CAMTNet's recognition accuracy improves by 1.3%, 2.2%, and 2.4% on the three irregular datasets, respectively. When compared to a model that solely relies on SPIN as its rectification module, CAMTNet achieves a 0.6% higher average recognition accuracy on irregular datasets. Moreover, the 1% improvement in average recognition accuracy over a model using only TPS underscores the rectification module's ability to effectively preprocess text images, thereby facilitating more accurate recognition.

Moreover, this section extracts visualizations of selected images from CUTE80, presented in Figure 12.



(a) Before

(b) After

Fig. 12. Irregular text images before and after correction.

Through visual comparison, the text in the rectified image is not only corrected to the horizontal position, but also more obvious to distinguish between the text and the background, and the characters in word images are clearer and easier to recognize.

### B. The effectiveness of feature extraction network

To validate the performance of our feature extraction network, TABLE VI and TABLE VII list the accuracy of three models based on VGGNet [30], VGGNet-CA, ResNet and ResNet-CA in regular and irregular datasets.

TABLE VI
Comparsion of VGGNet, VGGNet-CA and ResNet-CA in regular datasets

| Model | Accuracy(%) | | |
|---|---|---|---|
| | IC13 | IIIT5K | SVT |
| VGGNet | 93.2 | 95.5 | 88.7 |
| VGGNet-CA | 93.7 | 95.6 | 89.2 |
| ResNet-CA | **95.8** | **96.9** | **91.9** |

TABLE VI shows that, the integration of CA within the VGGNet architecture yields a 0.4% enhancement in average recognition accuracy on regular datasets. This modest yet significant gain can be attributed to the CA mechanism's ability to direct the network's focus towards the most informative regions of the input data, thereby enriching the feature representation. Furthermore, the substitution of VGGNet-CA with ResNet-CA procures an even more substantial improvement, with an average recognition accuracy increase of 2.0% on the same datasets. The experimental evidence reaffirms the efficacy of both CA and ResNet in enhancing the recognition of regular images. The CA mechanism's role in emphasizing relevant features and ResNet's proficiency in learning hierarchical feature representations contribute to CAMTNet's superior performance.

TABLE VII
Comparsion of VGGNet, VGGNet-CA and ResNet-CA in irregular datasets

| Model | Accuracy(%) | | |
|---|---|---|---|
| | SVTP | CT | IC15 |
| VGGNet | 81.3 | 84.9 | 80.4 |
| VGGNet-CA | 81.3 | 85.2 | 80.8 |
| ResNet-CA | **86.1** | **88.5** | **84.8** |

The results presented in TABLE VII are indicative of the significant enhancements achieved by CAMTNet in the realm of irregular image recognition. VGGNet-CA is improved by 0.2% in the irregular datasets, compared to the VGGNet. Meanwhile, the substitution of VGGNet-CA with ResNet-CA has yielded even more remarkable results, with an average recognition accuracy improvement of 4% on irregular datasets. Especially on SVTP, ResNet-CA accuracy increased by 4.8% compared to VGGNet-CA. These findings suggest that the use of a deeper convolutional network and CA can improve CAMTNet's recognition performance for irregular images.

### C. *The effectiveness of CA mechanism*

For confirming the impact of CA mechanism on the model, this section uses regular and irregular datasets for testing, and the results are shown in TABLE VIII and TABLE IX.

TABLE VIII
COMPARSION OF PERFORMANCE WITH OR WITHOUT CA IN REGULAR DATASETS

| Model | Accuracy(%) | | |
|---|---|---|---|
| | IC13 | IIIT5K | SVT |
| w/o CA | 95.4 | 96.7 | 90.2 |
| CA | **95.8** | **96.9** | **91.9** |

TABLE VIII compares the results in the regular datasets for the model with and without CA. It can be seen from the results that the average recognition accuracy of the model with CA is 0.8% higher than that of the model without CA. Especially for identifying SVT, add the CA model recognition accuracy rather than not adding 1.7%.

TABLE IX
COMPARSION OF PERFORMANCE WITH OR WITHOUT CA IN IRREGULAR DATASETS

| Model | Accuracy(%) | | |
|---|---|---|---|
| | SVTP | CT | IC15 |
| w/o CA | 85.4 | 86.8 | 81.4 |
| CA | **86.1** | **88.5** | **84.8** |

TABLE IX shows the experimental results of models with and without CA on irregular datasets. The average recognition accuracy of the model with CA is 1.9% higher than that of the model without CA. This validates that CA can enhance the model's performance, facilitating the feature extraction network in better capturing the features of the text.

Furthermore, to assess the influence of various attention mechanisms on model performance, this section conducts a comparative analysis of models utilizing two distinct attention mechanisms, namely Convolutional Block Attention Module (CBAM) [31] and CA, on six test datasets. The experimental outcomes are illustrated in Figure 13.

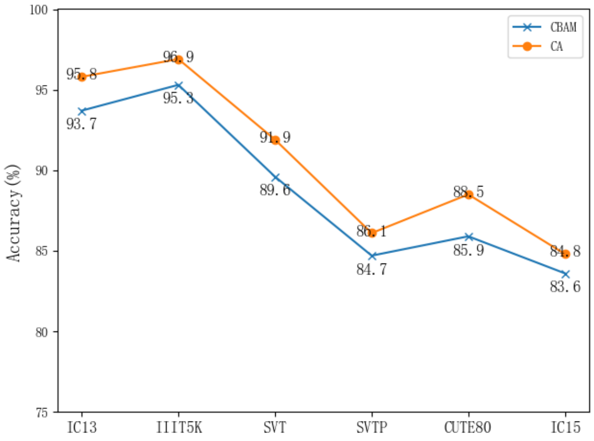


Fig. 13. Accuracy comparison of CBAM and CA.

The figure shows that compared with CBAM, the recognition accuracy of CA on test datasets is improved. This consistent outperformance of CA across diverse datasets underscores its superior capabilities.

### D. *The effectiveness of Transformer*

To evaluate the effectiveness of the Transformer, this section conducted experiments using models that employ BiLSTM, BiGRU, and Transformer as the encoder module, while keeping other components consistent. The results of these experiments are detailed in TABLE X and TABLE XI.

TABLE X
COMPARSION OF BLSTM, BIGRU AND TRANSFORMER IN REGULAR DATASETS

| Model | Accuracy(%) | | |
|---|---|---|---|
| | IC13 | IIIT5K | SVT |
| BiLSTM | 92.9 | 93.2 | 87.3 |
| BiGRU | 95.1 | 96.2 | **92.1** |
| Transformer | **95.8** | **96.9** | 91.9 |

According to the experimental results in TABLE X, different encoder modules will affect the recognition effect of the model. The average recognition accuracy of models using BiLSTM, BiGRU, and Transformer in the irregular datasets is 91.1%, 94.5%, and 94.9%, respectively. The average recognition accuracy of the model using Transformer is 3.8% and 0.4% higher than the first two models, respectively. Models using Transformer show better recognition performance.

TABLE XI
COMPARSION OF BLSTM, BIGRU AND TRANSFORMER IN IRREGULAR DATASETS

| Model | Accuracy(%) | | |
|---|---|---|---|
| | SVTP | CT | IC15 |
| BiLSTM | 82.8 | 84.2 | 78.1 |
| BiGRU | 84.3 | 85.5 | 82.3 |
| Transformer | **86.1** | **88.5** | **84.8** |

TABLE XI represents the recognition results in irregular datasets using different encoder modules. According to the data in the table, the average recognition accuracy of the model using Transformer is 86.5%, which is 4.8% and 2.5% higher than that of the model using BiLSTM and BiGRU, respectively. The experimental results further verify the effectiveness of Transformer as an encoder module.

In addition, to make the experimental results clearer, this section visualizes the recognition results of models with three different encoder modules, as shown in Figure 14.

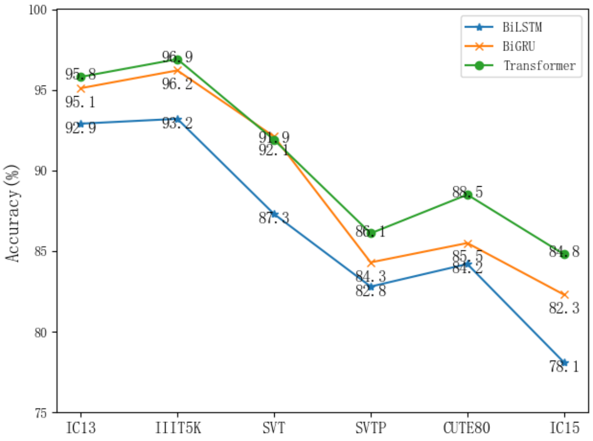


Fig. 14. Accuracy comparison of BiLSTM, BiGRU and Transformer.

From Figure 14, it is clear that on six different datasets, the model using BiGRU achieves higher recognition accuracy than the model employing BiLSTM. Moreover, the

Transformer-based model demonstrates a higher recognition advantage than the other two models.

### *E. The influence of hyperparameter α*

In the decoder module, a hyperparameter α is introduced to adjust the balance between the CTC and the sequence-based attention mechanism. To determine the optimal value, this section tests the effect on the model when α is set to different values on three irregular datasets. In addition, to expedite training, the initial 500,000 images from MJ are used to train the model, with the experimental results shown in TABLE XII.

TABLE XII
THE INFLUENCE OF DIFFERENT α VALUE
IN REGULAR DATASETS

| α | Accuracy(%) | | | | | |
|---|---|---|---|---|---|---|
| | IC13 | IIIT5K | SVT | SVTP | CT | IC15 |
| 0.3 | 75.8 | 76.7 | 75.0 | 55.7 | 53.8 | 51.9 |
| 0.4 | 76.4 | 76.9 | 76.7 | 54.9 | 54.2 | 51.1 |
| 0.5 | 76.0 | 78.0 | 75.9 | 56.8 | 55.7 | 52.2 |
| 0.6 | 76.4 | 77.1 | 76.9 | 55.4 | 57.3 | 52.3 |
| 0.7 | 75.2 | 77.1 | 76.5 | 56.1 | 50.7 | 52.7 |
| 0.8 | 75.7 | 77.7 | 75.2 | 55.7 | 52.8 | 52.3 |
| 0.9 | 75.4 | 78.3 | 75.4 | 53.0 | 54.9 | 51.2 |

Table XII presents a pivotal analysis showcasing the impact of the hyperparameter α on CAMTNet's recognition accuracy. It is observed that α plays a critical role in balancing the contributions of the CTC and the sequence-based attention mechanism within the decoder module. As α increases, the model's recognition accuracy initially improves, reflecting the enhanced synergy between the CTC and attention mechanisms. However, beyond a certain threshold, the accuracy begins to decline, indicating that the balance has tipped too far towards one mechanism over the other, potentially leading to an overemphasis on certain features at the expense of others. The optimal value of α is identified to be 0.6, at which the model achieves its peak performance, with the highest overall recognition accuracy.

## VI. CONCLUSION

This paper proposes a scene text recognizer based on CRNN, and named as CAMTNet. It attains an average recognition accuracy of 94.9% on regular datasets and 86.5% on irregular datasets, showcasing a 7% and 20% improvement over the baseline CRNN. Furthermore, compared to other text recognition models, the overall recognition performance reaches an optimum level. However, the model's encoder module uses the basic Transformer, which has numerous parameters and is unsuitable for low-configuration devices. In the future, we will continue to optimize the text recognition model in this paper, and further study the recognition of low-quality word images, while using a lightweight encoder module.

## REFERENCES

[1] Y. H. Cheng, P. C. Chang, D. M. Nguyen, and Kuo, C. N, "Automatic Music Genre Classification Based on CRNN," Engineering Letters, vol.29, no.1, pp312-316, 2021

[2] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," IEEE Trans. Pattern Anal. Mach. Intell, vol.37, no.7, pp1480-1500, 2015

[3] B. Shi, X. Bai, C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," IEEE Trans. Pattern Anal. Mach. Intell, vol.39, no.11, pp2298-2304, 2016

[4] H. Pan, W. Huang, Q. Yu, C. Loy, and X. Tang, "Reading scene text in deep convolutional sequences," In Proceedings of the AAAI Conference on Artificial Intelligence, vol.30, no.1, pp254-283, 2016

[5] C. Zhang, Y. Xu, Z. Cheng, S. Pu, Y. Niu, F. Wu, and F. Zou, "Spin: Structure-preserving inner offset network for scene text recognition," In Proceedings of the AAAI Conference on Artificial Intelligence, vol.35, no.4, pp3305-3314, May 2021

[6] A. Vaswani, N. Shazeer, N. Parmar, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, "Attention is all you need," Advances in Neural Information Processing Systems, vol.30, 2017

[7] C. Luo, L. Jin, and Z. Sun, "MORAN: A multi-object rectified attention network for scene text recognition," Pattern Recognition, vol.90, pp109-118, 2019

[8] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai, "Robust scene text recognition with automatic rectification," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp4168-4176, 2016

[9] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," Advances in Neural Information Processing Systems, vol.28, pp2017-2025, 2015

[10] H. Heng, P. Li, T. Guan, and T. Yang, "Scene text recognition via context modeling for low-quality image in logistics industry," Complex & Intelligent Systems, pp1-20, 2022

[11] J. Chen, H. Yu, J. Ma, B. Li, and X. Xue, "Text Gestalt: Stroke-Aware Scene Text Image Super-Resolution," Proc. the AAAI Conference on Artificial Intelligence, vol.36, no.1, pp285-293, June 2021

[12] Y. L. Tan, E. Y. K. Chew, A. W. K. Kong, J. J. Kim and J. H. Lim, "Portmanteauing Features for Scene Text Recognition," In 2022 26th International Conference on Pattern Recognition(ICPR), IEEE, Montreal, Canada, pp1499-1505, August 2022

[13] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao and X. Bai, "Aster: an attentional scene text recognizer with flexible rectification," IEEE Trans Pattern Anal Mach Intell, vol.41, no.9, pp2035-2048, 2019

[14] N. Lu, W. Yu, X. Qi, Y. Chen, P. Gong, R. Xiao and X. Bai, "Master: Multi-aspect non-local network for scene text recognition," Pattern Recognition, vol.117, pp107980, 2021

[15] Y. Du, Z. Chen, C. Jia, X. Yin, T. Zheng, C. Li, Y. Du, and Y. G. Jiang, "Svtr: Scene Text Recognition with a Single Visual Model," arXiv preprint arXiv: 2205.00159, 2022

[16] W. Hu, X. Cai, J. Hou, S. Yi, and Z. Lin, "Gtc: Guided training of ctc towards efficient and accurate scene text recognitiona," In Proceedings of the AAAI Conference on Artificial Intelligence, vol.34, no.7, pp11005-11012, April 2020

[17] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," arXiv preprint arXiv: 1406.2227, 2014

[18] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp2315-2324, 2016

[19] A. Mishra, K. Alahari, C. Jawahar, "Scene text recognition using higher order language priors," In BMVC-British Machine Vision Conference, BMVA, 2012

[20] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan,and L. P. de las Heras, "ICDAR 2013 robust reading competition," In 2013 12th International Conference on Document Analysis and Recognition, IEEE, pp1484-1493, August 2013

[21] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," In 2011 International Conference on Computer Vision, IEEE, pp1457-1464, Nov. 2011

[22] T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan, "Recognizing text with perspective distortion in natural scenes," In Proceedings of the IEEE International Conference on Computer Vision, IEEE, pp569-576, 2013

[23] A. Risnumawan, P. Shivakumara, C. S. Chan, and C. L. Tan, "A robust arbitrary text detection system for natural scene images," Expert Systems with Application, vol.41, no.18, pp8027-8048

[24] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny, "ICDAR 2015 competition on robust reading," In 2015 13th International Conference on Document Analysis and Recognition(ICDAR), IEEE, pp1156-1160, August 2015

[25] J. Baek, G. Kim, J. Lee, S. Park, D. Han, S. Yun, S. J. Oh, and H. Lee, "What is wrong with scene text recognition model comparisons? dataset and model analysis," In Proceedings of the IEEE/CVF International Conference on Computer Vision, IEEE, pp4715-4723, 2019

[26] D. Yu, X. Li, C. Zhang, T. Liu, J. Han, J. Liu, and E. Ding, "Towards accurate scene text recognition with semantic reasoning networks," In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, pp12113-12122, 2020

[27] Y. Wang, H. Xie, S. Fang, J. Wang, S. Zhu, and Y. Zhang, "From two to one: A new scene text recognizer with visual language modeling network," In Proceedings of the IEEE/CVF International Conference on Computer Vision, IEEE, pp14194-14203, 2021

[28] C. Liu, C. Yang, and X. Yin, "Open-Set Text Recognition via Character-Context Decoupling," In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, pp4523-4532, 2022

[29] H. Yue, Y. Huang, C. M. Vong, Y. Jin, Z. Zeng, M. Yu, and C Chen, "NRSTRNet: A Novel Network for Noise-Robust Scene Text Recognition," International Journal of Computational Intelligence Systems, vol.16, no.1, pp5, 2023

[30] L. Cui, and Y. Tian, "Facial Expression Recognition by Regional Attention and Multi-task Learning," Engineering Letters, vol.29, no.3, pp919-925, 2021

[31] X. Kuang, P. Liu, Y. Chen, and J. Zhang, "Lightweight Semantic Segmentation Network based on Attention Feature Fusion," Engineering Letters, vol.31, no.4, pp1584-1591, 2023