

Article

Character-Level Visual Guidance for Arbitrarily Shaped Scene Text Detection and Recognition

Lijia Chen ¹, Hu Lin ² and Dan Chen ^{2,*}

¹ Department of Environmental Design, Fujian Business University, Fuzhou 350012, China; chenlijia@fjbu.edu.cn

² Department of Information Engineering, Fuzhou University, Fuzhou 350001, China; 2501110034@fzu.edu.cn

* Correspondence: cd@fzu.edu.cn

Abstract

Curved, blurred, low-contrast, and cluttered scene text remains difficult to localize and transcribe because visual boundaries and character identities are often ambiguous. Most detectors refine spatial regions from visual and positional features, while many recognizers depend on implicit attention between global image tokens and language context. This separation weakens character-level evidence in both localization and text decoding. To address this issue, this paper studies explicit character-level guidance for scene text detection and recognition. For arbitrary-shaped text detection, the character-guided adaptive detector (CADet) combines a text-enhancement network (TENet), a character information adaptive guidance module (CIA), and a position and classification compensation module (COMP). Then, character semantics can participate in boundary-query construction before final detection. For scene text recognition (STR), serialized image embeddings for text recognition (SIETR) introduce local visual embeddings into autoregressive decoding via the character local image embedding module (CLIE) and permutation language modeling (PLM). On ArT, Total-Text, and CTW1500, CADet obtains F-measures of 79.5%, 89.4%, and 89.2%, respectively, outperforming representative Transformer-based detectors with only a small increase in computation. For recognition, SIETR reaches 95.6% sample-size-weighted average accuracy with 23.8 M parameters and 3.2 G FLOPs and improves most irregular-text benchmarks over PARSeq with fewer FLOPs. The results indicate that character semantics for detection and local visual evidence for decoding are an effective way to improve robustness in difficult scene text.

Keywords: arbitrary-shaped text detection; scene text recognition; explicit guidance; character-level semantics; visual–context alignment



Academic Editor: George Angelos Papadopoulos

Received: 3 August 2026

Revised: 8 September 2026

Accepted: 9 September 2026

Published: 11 September 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 \(CC BY\) license](#).

1. Introduction

Text in natural images conveys rich spatial, visual, and semantic information and plays an important role in applications such as intelligent transportation, mobile translation, information retrieval, and human–computer interaction [1,2]. Compared with document optical character recognition (OCR), scene text analysis is much more challenging because text may appear against cluttered backgrounds and exhibit perspective distortion, curved layouts, uneven illumination, occlusion, low resolution, and substantial variations in font, scale, and orientation. These complex visual conditions make scene text analysis dependent on both accurate spatial localization and reliable semantic interpretation, rather than on geometric or linguistic modeling alone.

Deep learning has significantly advanced scene text detection and recognition. For detection, segmentation-based methods predict dense text maps and reconstruct text instances at the pixel level, improving the localization of arbitrary-shaped text [1]. Residual networks and Transformer architectures further enhance multi-scale feature representation and long-range dependency modeling [3–5], while deformable attention and focal loss improve query-based localization and dense classification [6,7]. For recognition, convolutional recurrent networks, attention-based decoders, language-enhanced models, and Vision Transformer architectures have achieved substantial performance improvements on both regular and irregular text benchmarks. Moreover, recent studies have extended scene text processing from separate detection and recognition toward end-to-end scene text spotting [2].

Despite recent progress, character-level visual evidence remains insufficiently exploited in both localization and transcription. Most detectors construct text regions or boundary queries mainly from visual and positional features without explicit character semantics, making them vulnerable to background textures and weak boundaries in curved, blurred, or low-contrast scenes. Meanwhile, many recognizers rely on global visual tokens and contextual modeling, where the correspondence between predicted characters and local image regions is only implicitly established. This weak association may lead to excessive dependence on global representations or language priors for distorted or ambiguous characters. Therefore, explicit character-level guidance can enhance semantic-to-geometric interaction in detection and local visual-to-sequential alignment in recognition.

To address these issues, this paper presents a two-branch character-level explicit guidance framework for arbitrary-shaped text detection and recognition (STR). CADet introduces character semantics into boundary-query construction, while SIETR incorporates character-local visual evidence into sequence decoding. The proposed CADet integrates TENet, CIA, and COMP to enhance text representations, guide boundary prediction, and refine localization. The proposed SIETR introduces local visual embeddings into autoregressive recognition through CLIE and joint decoding. Experiments show that the framework achieves F-measures of 79.5%, 89.4%, and 89.2% on ArT, Total-Text, and CTW1500, respectively, with limited computational overhead. Moreover, SIETR obtains 95.6% sample-size-weighted average accuracy with 23.8 M parameters and 3.2 G FLOPs, outperforming PARSeq on most irregular benchmarks.

2. Related Work

This section reviews representative methods of scene text detection and recognition. It first summarizes scene text detection methods from segmentation- and regression-based approaches to Transformer-based detectors, and then discusses scene text recognition methods with emphasis on visual-semantic alignment and local visual evidence.

2.1. Scene Text Detection

Scene text detection methods are commonly classified into regression-based and segmentation-based approaches. Regression-based methods locate text using rectangles, quadrangles, polygons, or control points. Segmentation-based methods predict dense text maps and were later used to describe arbitrary-shaped text regions. TextSnake and pixel aggregation networks improved arbitrary-shaped text detection by representing curved text through center lines, kernels, or progressive grouping [8,9]. Polygon offsetting and border-aware segmentation have also been used to separate adjacent arbitrary-shaped text instances [10]. Character Region Awareness for Text Detection (CRAFT) used character-region affinity to connect local text components [11]. Differentiable Binarization (DB) simplified post-processing [12], and Adaptive Bezier-Curve Network (ABCNet) repre-

sented curved text using Bezier curves [13]. These methods established strong foundations for irregular text localization, but dense prediction and post-processing can still be sensitive to background noise, adjacent text instances, and highly variable text scales. Recent advances in multi-modal representation learning have demonstrated the effectiveness of aligning visual information with semantic representations for robust perception [14]. This method improves feature representation and semantic understanding in complex scenarios that provide valuable inspiration for scene text analysis.

Transformer-based detectors reformulated text detection as query-based prediction. Text Spotting Transformer (TESTR) introduced Transformer representations into text spotting [15], while Dynamic Points Text Detection Transformer (DPTText-DETR) used explicit point queries to model arbitrary-shaped text instances [16]. SwinTextSpotter explored detection-recognition synergy, and SwinTextSpotter v2 further proposed recognition-guided localization. Recent StarNet-Tex proposed a multi-scale feature enhancement module for lightweight text detection [17–19]. Style-aware learning has also been introduced to improve detection robustness under appearance variation [20]. These methods improved global context modeling and reduced hand-designed post-processing. Nevertheless, most query-based detectors still depended mainly on location and visual features when selecting text candidates. Recognition-level character information was not sufficiently used as an explicit constraint during detection. This limitation can lead to false positives in background regions with text-like texture and missed detections for weak or curved text.

2.2. Scene Text Recognition

Scene text recognition (STR) transcribes cropped text images into character sequences. A convolutional recurrent neural network (CRNN) combined convolutional features with recurrent sequence modeling and connectionist temporal classification [21]. Later systems introduced attention mechanisms, rectification modules, bidirectional semantic modeling, and language priors. Vision Transformer for Scene Text Recognition (ViTSTR) represented text images as token sequences and achieved efficient recognition [22]. Autonomous Bidirectional and Iterative Network (ABINet) explicitly modeled language information [23], and permuted autoregressive sequence modeling (PARSeq) reduced dependence on a single left-to-right decoding order [24]. Recent recognition studies further show that performance in complex scenes depends on robust visual-semantic alignment, CTC-compatible feature alignment, multi-scale attention, and controlled integration of language cues [25–28].

Despite these advances, sequence decoders often depend on global visual tokens and implicit cross-modal attention. This design can be insufficient when adjacent characters are distorted, occluded, or visually similar. Contextual semantics can correct some errors, but they can also introduce visually unsupported predictions. A robust recognizer therefore needs sequence context and explicit local visual information associated with each predicted character. SIETR follows this direction by embedding serialized local image features into the character prediction process. Taken together, these studies confirm that practical OCR systems increasingly require interaction between detection, recognition, restoration, and language constraints rather than isolated modules.

3. Materials and Methods

This section describes the proposed explicit-guidance framework. It first formulates the detection and recognition tasks, then presents the overall framework, the CADet detection branch, the SIETR recognition branch, and the datasets and evaluation metrics used for validation.

3.1. Problem Formulation

Given an input scene image I , scene text detection estimates a set of text instances

$$\mathcal{B} = \{(b_i, c_i)\}_{i=1}^N, \quad (1)$$

where b_i denotes the boundary representation of the i th text instance and c_i denotes its text confidence. STR then predicts a character sequence

$$\hat{Y} = (\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T), \quad (2)$$

for each detected or cropped text image. The paper studies two aligned tasks, i.e., robust arbitrary-shaped text localization and accurate sequence recognition. The methodological assumption is that detection benefits from explicit character semantics, whereas recognition benefits from explicit character-local image information.

3.2. Overall Framework

Figure 1 summarizes the proposed explicit-guidance strategy. The framework is organized as two branches for a detection-then-recognition scene text analysis pipeline. Its connection lies in the shared explicit character-level guidance principle: semantic information guides spatial localization in CADet, while local visual information grounds sequence prediction in SIETR. The detection branch uses text-enhanced features, boundary sampling points, character information adaptive guidance, and compensation modules to output Bezier-based instances. The recognition branch converts an input word image into visual tokens, learns contextual dependencies with permuted attention masks, embeds local image features into position queries, and predicts the final sequence through a joint decoder.

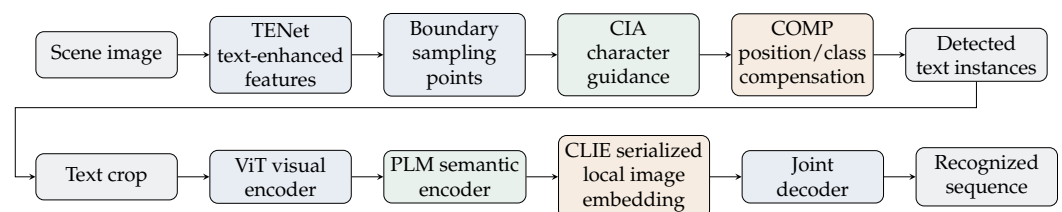


Figure 1. Explicit character-level visual-context guidance used in this paper. The upper branch corresponds to CADet for arbitrary-shaped text detection, and the lower branch corresponds to SIETR for STR.

The proposed framework introduces explicit character-level information at key stages of detection and recognition. In the detection branch, character semantics are incorporated before boundary refinement, helping suppress text-like background patterns and recover weak text instances. In the recognition branch, local image embeddings are introduced before decoding to maintain the correspondence between predictions and character-level visual evidence. Unlike post hoc language correction or purely visual boundary regression, the proposed methods directly constrain recognition with local visual cues and detection with character semantics.

The framework investigates independent information pathways rather than simply stacking modules. Specifically, TENet enhances text-region representations, CIA integrates character semantics into boundary points, COMP refines localization and classification errors, and CLIE provides local visual evidence for decoding. Each module targets a specific challenge in scene text analysis and enables interpretable ablation analysis.

Figure 2 expands the upper branch of Figure 1. It shows that CADet does not append a recognition cue after detection. Instead, text-enhanced features, sampled boundary points, character guidance, and compensation are arranged before the final Transformer

decoding stage. This order is central to the hypothesis tested later: character-level evidence should affect candidate localization while the detector is still forming its spatial query representation.

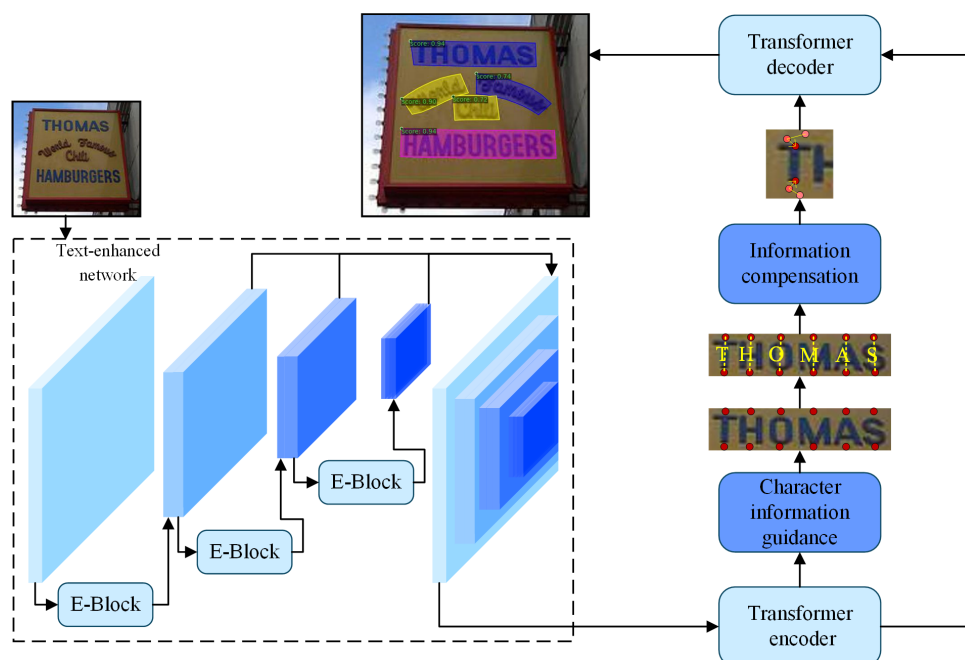


Figure 2. Detailed architecture of CADet for arbitrary-shaped text detection. The detector combines text-enhanced visual features, boundary-point queries, character-information guidance, and residual compensation before instance prediction.

3.3. Character-Guided Adaptive Text Detection

CADet is built on a Transformer-based arbitrary-shaped text detector. The backbone is TENet, which uses ResNet-50 with text-enhancement blocks to strengthen text-region features and suppress background interference. Multi-scale features are concatenated and passed to a deformable-attention Transformer encoder:

$$F_e = \text{Encoder}(\text{Concat}(F_1, F_2, F_3) + P_s), \quad (3)$$

where $F_l \in \mathbb{R}^{H_l W_l \times d}$ denotes the l -th scale feature after channel projection and spatial flattening, with H_l and W_l being the height and width of the l -th feature map and $d = 256$ being the hidden dimension. Thus, $\text{Concat}(F_1, F_2, F_3) \in \mathbb{R}^{(\sum_{l=1}^3 H_l W_l) \times d}$ is obtained by concatenating the three feature sequences along the spatial-token dimension. The sinusoidal positional encoding P_s is generated at the corresponding spatial locations and has the same dimension as the concatenated feature sequence, enabling element-wise addition before the deformable-attention Transformer encoder. For each prior text box, the upper and lower boundaries are sampled into control points. These points provide Bezier-style geometric priors for arbitrary-shaped text.

The CIA module introduces recognition-level character semantics into candidate queries. For each sampled point, classification features are projected to a probability vector over 26 letters and one non-character class:

$$p_i = \text{Softmax}(W_2 \sigma(W_1 f_i + b_1) + b_2), \quad (4)$$

where f_i is the encoded feature at the sampled point and $\sigma(\cdot)$ denotes ReLU activation. The vector p_i serves as a soft character-semantic descriptor inferred from the sampled-point feature. It encodes alphabetic and non-character responses around the sampled position.

After projection by $\phi_{\text{char}}(\cdot)$ and adaptive weighting by λ_i , this descriptor is combined with the positional encoding of the sampled control point to form q_i . Therefore, each explicit point-composite query carries both a Bezier-style geometric prior and local character/non-character cues, allowing CIA to distinguish character-like sampled points from background or blank regions before boundary decoding. The semantic vector is fused with positional information to form an explicit point-composite query:

$$q_i = \phi_{\text{pos}}(x_i, y_i) + \lambda_i \phi_{\text{char}}(p_i), \quad (5)$$

where λ_i denotes the point-wise character-confidence weight generated by the CIA sub-network for the i -th sampled point. It is computed from the sampled-point feature f_i through the character-information branch and scales $\phi_{\text{char}}(p_i)$ according to the predicted text/character confidence. This query guides the detector to select candidates that are geometrically plausible and semantically text-like.

During training, candidate text instances are assigned to ground-truth text instances by Hungarian matching. The optimization target is the matched text instance, including text confidence and control-point geometry. The CIA branch is optimized through gradient backpropagation of the decoder from the matched-instance detection losses.

The operation in Equation (5) performs explicit semantic guidance rather than generic feature fusion. As shown in Figure 3, the sampled text-region features are first processed by the character-information branch and then transformed into a composite query. The non-character channel represents non-text or blank responses around the sampled position, while the character probability distribution provides fine-grained semantic discrimination. By retaining weak but character-consistent candidate points, CIA can improve text-instance recovery and therefore has a greater influence on recall than on precision in the ablation study.

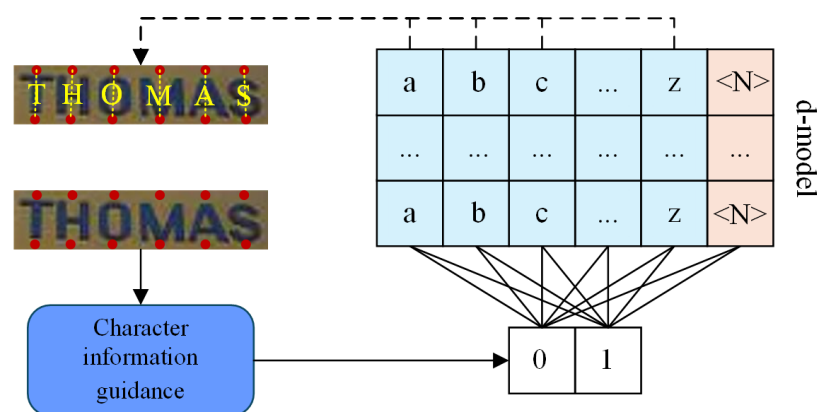


Figure 3. Character information adaptive guidance in the detection branch. Character probabilities are used to form point-composite queries that connect geometric sampling with recognition-level evidence.

The COMP module corrects the spatial offset and classification confidence of the explicit query. Two zero-initialized lightweight multi-layer perceptrons (MLPs) estimate light residual compensation:

$$\tilde{q}_i = q_i + \Delta q_i^{\text{pos}} + \Delta q_i^{\text{cls}}. \quad (6)$$

The COMP module is used as residual calibration for the explicit point-composite query. Although q_i combines sampled control-point geometry and character-semantic evidence, its initial sampled position and classification confidence can still be biased by cluttered background textures or weak text contrast [2,16]. Therefore, following the residual-learning principle and the query-refinement [4,6,16], COMP uses two lightweight MLPs as residual

prediction functions to estimate Δq_i^{pos} and Δq_i^{cls} . These two terms compensate position-related and classification-related query errors before decoding. The residual outputs are initialized to zero, so $\tilde{q}_i = q_i$ at the beginning of training. Since the residual outputs are initialized to zero, the compensated query is identical to the original query at the start of training, i.e., $\tilde{q}_i = q_i$. Thus, COMP does not disturb the original detection query at initialization. During training, the two MLPs, as a query-calibration mechanism, gradually learn only the residual adjustments needed to reduce the matched-instance detection losses. The final detection loss combines focal classification loss, point regression loss, and GIoU supervision:

$$\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{cls}} + \alpha \mathcal{L}_{\text{point}} + \beta \mathcal{L}_{\text{giou}}. \quad (7)$$

3.4. Serialized Image Embeddings for Text Recognition

SIETR is designed to strengthen the association between each predicted character and its corresponding image region. The visual encoder follows a ViT-style architecture. Given an input word image $I_w \in \mathbb{R}^{H \times W \times C}$, the image is divided into patches and flattened into visual tokens:

$$N_p = \frac{H \times W}{P^2}, \quad F_v = \text{ViT}(\text{PatchEmbed}(I_w) + P_v), \quad (8)$$

where P is the patch size, N_p is the number of patches, and P_v is learnable positional encoding.

The semantic encoder uses PLM to reduce dependence on a single left-to-right order. The conditional probability of a sequence is written as

$$p(Y|I_w) = \prod_{t=1}^T p(y_t|y_{<t}, I_w), \quad (9)$$

while the permuted formulation approximates multiple factorization orders:

$$\mathcal{L}_{\text{plm}} = -\mathbb{E}_{z \sim \mathcal{Z}_T} \sum_{t=1}^T \log p(y_{z_t}|y_{z_{<t}}, I_w). \quad (10)$$

In practice, only a limited number of permutations is sampled to avoid factorial growth. The experiments indicate that six permutation masks provide the best overall trade-off.

The CLIE module converts visual tokens into serialized local image embeddings. Visual features are projected into a character-related feature space, top- K_{loc} local responses are selected, and the selected features are normalized:

$$G = \text{Linear}(F_v), \quad E = [E_1, \dots, E_T] = \text{Norm}(\text{TopK}(G, K_{\text{loc}})), \quad K_{\text{loc}} = T. \quad (11)$$

Here, K_{loc} denotes the number of retained local visual responses in CLIE. The TopK operation is performed within each cropped word image: after the linear projection, the visual tokens with the highest character-response confidence are retained and normalized as serialized local image embeddings.

During training, we set $K_{\text{loc}} = T$ to select character-related local visual embeddings of each target position. The target sequence length T is available from the ground-truth transcription. During inference, since the target length is unknown before decoding, K_{loc} is set to the predefined maximum recognition length T_{max} . In our experiments, T_{max} is set to 25, following common scene text recognition methods [23]. The decoder predicts characters autoregressively and stops when the EOS token is generated. Local embeddings corresponding to positions after EOS are ignored. The semantic query s_t and local image embedding E_t are fused as

$$\tilde{s}_t = s_t + E_t. \quad (12)$$

Equation (12) provides a learned soft-alignment mechanism rather than a hard coordinate assignment. Since F_v is generated from image patches with learnable positional encoding P_v , the selected local responses retain spatial context. After E_t is added to the semantic query s_t , the fused query $\tilde{s}_t$ attends to the visual-token sequence in the joint decoder. The recognition loss at each character position then guides the decoder to associate the t -th query with its relevant local visual region. Therefore, CLIE introduces character-local visual evidence without requiring character-level bounding boxes or explicit coordinate constraints. The joint decoder predicts the output character distribution:

$$\hat{Y} = \text{Softmax}(W_o \text{Decoder}(\tilde{S}, F_v, M_{ar}) + b_o), \quad (13)$$

where M_{ar} is the autoregressive attention mask. The fused query sequence $\tilde{S}$ is denoted as $\tilde{S} = \{\tilde{s}_1, \tilde{s}_2, \dots, \tilde{s}_T\}$, where each fused query $\tilde{s}_t$ combines semantic context and character-local visual evidence as in (12). The recognition loss is the mean cross-entropy between the predicted and ground-truth character sequences:

$$\mathcal{L}_{rec} = -\frac{1}{T} \sum_{t=1}^T \log p(y_t | I_w). \quad (14)$$

Figure 4 clarifies the role of CLIE within the recognition branch. The semantic encoder supplies contextual character queries, but the serialized image embedding module reintroduces local image evidence before joint decoding. Thus, the decoder receives both a context-conditioned sequence representation and local visual features. This placement is important for irregular text, where a plausible language sequence may conflict with the actual distorted character image.

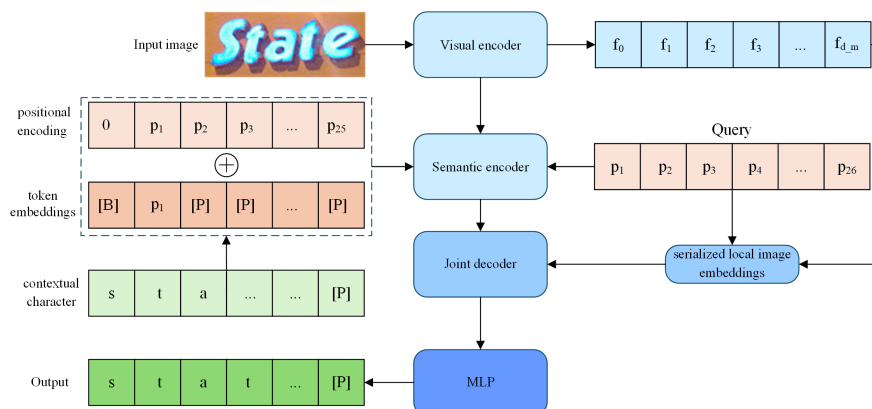


Figure 4. Detailed architecture of SIETR for STR. The model links ViT-based visual tokens, permutation-based semantic encoding, serialized local image embedding, and joint decoding.

3.5. Datasets and Evaluation Metrics

The detection experiments used ArT, Total-Text, and CTW1500 [29–31]. ArT contains 10,166 arbitrary-shaped text images, with 5603 training images and 4563 test images. Total-Text contains 1255 training images and 300 test images with horizontal, multi-oriented, and curved text. CTW1500 contains 1000 training images and 500 test images, with long curved text annotated by 14-point polygons. Additional pretraining used Synth150K, ICDAR2017 MLT, and ICDAR2019 LSVT following the same protocol for compared models.

The recognition experiments used nine training datasets, including COCO-Text, RCTW17, Uber-Text, ArT, LSVT, MLT19, ReCTS, TextOCR, and OpenVINO. Evaluation was conducted on IIIT5K, SVT, SVTP, CUTE, ICDAR2013, and ICDAR2015 [32–36]. IIIT5K, SVT, and ICDAR2013 represent relatively regular text, while SVTP, CUTE, and ICDAR2015

emphasize perspective distortion, curved text, motion blur, low resolution, and irregular layouts.

The experimental design uses three controls. First, detection comparisons use the same public benchmark splits and the same precision, recall, and F-measure metrics. Second, recognition comparisons report regular and irregular datasets separately, because the proposed local visual embedding is expected to matter more under distortion. Third, parameter and FLOP comparisons are reported with accuracy, so gains can be judged against computational cost rather than accuracy alone.

Detection is evaluated by precision, recall, and F-measure:

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F = \frac{2PR}{P + R}. \quad (15)$$

Here, TP , FP , and FN denote true positives, false positives, and false negatives, respectively. The reported precision and recall are rounded to one decimal place, and the calculated F-measure is rounded accordingly. Recognition is evaluated by exact-match sequence accuracy:

$$Acc = \frac{N_{correct}}{N_{correct} + N_{wrong}}. \quad (16)$$

4. Results

This section reports the experimental results. It first analyzes the contribution of each detection and recognition component through ablation studies, and then compares CADet and SIETR with representative methods on public scene text benchmarks.

4.1. Ablation Analysis for Detection

Table 1 reports the contribution of TENet, CIA, and COMP on ArT and Total-Text. On ArT, replacing ResNet-50 with TENet increased the F-measure from 78.1% to 78.3%. Adding CIA alone raised recall to 75.6%, indicating that recognition-level character semantics helped recover missed text instances. Adding COMP alone improved the F-measure to 78.9%. The complete CADet model achieved the best F-measure of 79.5%. On Total-Text, the complete model achieved the best F-measure of 89.4%, mainly because recall reached 88.0%.

Table 1. Ablation results for CADet on ArT and Total-Text.

Dataset	TENet	CIA	COMP	P	R	F
ArT	–	–	–	83.4	73.4	78.1
ArT	✓	–	–	83.4	73.8	78.3
ArT	–	✓	–	81.9	75.6	78.6
ArT	–	–	✓	82.7	75.4	78.9
ArT	✓	✓	–	82.4	75.6	78.9
ArT	✓	–	✓	83.5	74.5	78.7
ArT	✓	✓	✓	83.3	76.0	79.5
Total-Text	–	–	–	92.3	86.0	89.0
Total-Text	✓	–	–	90.3	88.0	89.1
Total-Text	–	✓	–	91.5	86.7	89.0
Total-Text	–	–	✓	91.4	86.8	89.0
Total-Text	✓	✓	–	90.0	88.4	89.2
Total-Text	✓	–	✓	90.7	87.8	89.2
Total-Text	✓	✓	✓	90.9	88.0	89.4

On ArT, TENet+CIA increased recall from 73.8% to 75.6%, whereas TENet+COMP produced a balanced result of 83.5%/74.5%/78.7% in P/R/F, supporting the recall-oriented role of character guidance and the residual-refinement role of COMP. On Total-Text, TENet+CIA raised recall from 88.0% to 88.4%, while TENet+COMP increased precision from 90.3% to 90.7%; both variants reached 89.2% F-measure, and the full model further improved to 89.4%, indicating that the CIA and COMP are complementary.

The ablation pattern shows that the detector gain does not come from a single stronger backbone. On ArT, CIA produced the largest recall increase among the single-module variants, from 73.4% to 75.6%. COMP produced a similar F-measure gain while keeping precision above 82%. The full model combined these effects and improved the baseline F-measure by 1.4 percentage points on ArT. On Total-Text, TENet and the full model both raised recall to 88.0%, while the full model recovered the best F-measure through a better balance between precision and recall.

The feature maps in Figure 5 provide qualitative support for the TENet component. Compared with the ResNet-50, TENet concentrates activation on text strokes and suppresses non-text background regions around the sign. This visual pattern is consistent with the recall increase in Table 1, because stronger text-region activation gives the subsequent boundary sampling and character-guidance modules a cleaner candidate set.

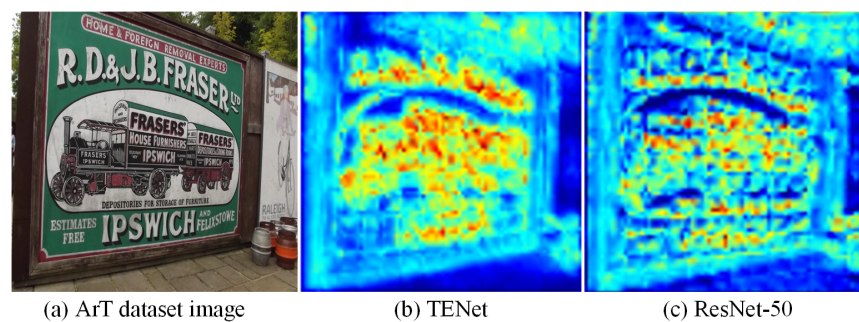


Figure 5. Feature visualization of TENet and ResNet-50. TENet produces stronger text-region responses, which supports the recall improvement observed in the detection ablation.

4.2. Comparison with Scene Text Detectors

Table 2 compares CADet with representative detection methods. On ArT, CADet achieved a 79.5% F-measure, which was 3.7 percentage points higher than TextBPN++ and 1.4 percentage points higher than DPText-DETR. On Total-Text, CADet obtained an 89.4% F-measure, improving over DPText-DETR by 0.4 percentage points and over SwinTextSpotter by 1.4 percentage points. Compared with recent SwinTextSpotter v2 and StarNet-Tex, CADet achieved gains of 1.7 and 2.9 percentage points on Total-Text, respectively. On CTW1500, CADet achieved an 89.2% F-measure, improving over DPText-DETR by 0.3 percentage points while achieving a gain of 4.1 percentage points compared to the StarNet-Tex. These results show that CADet injects character semantics into boundary-point query construction and applies residual query compensation before decoding, which helps preserve weak and curved text boundaries.

The computational comparison shows that CADet remains within the same parameter and FLOP range as DPText-DETR. CADet uses 45.9 M parameters and 284.5 G FLOPs, compared with 45.8 M parameters and 279.8 G FLOPs for DPText-DETR. The performance gain is therefore not obtained by a large increase in model size.

This comparison is important for deployment-oriented evaluation. The additional character guidance increases computation by 4.7 G FLOPs relative to DPText-DETR, which is about 1.7% of the DPText-DETR cost. The improvement therefore reflects a more informative query representation rather than a substantially larger detector.

Table 2. Detection comparison on arbitrary-shaped text benchmarks.

Model	ArT			Total-Text			CTW1500		
	P	R	F	P	R	F	P	R	F
TextSnake	–	–	–	82.7	74.5	78.4	67.9	85.3	75.6
CRAFTS	77.2	68.9	72.8	87.6	79.9	83.6	86.0	81.1	83.5
TextFuseNet	82.6	69.4	75.4	87.5	83.2	85.3	85.8	85.0	85.4
TextBPN++	81.1	71.1	75.8	91.8	85.3	88.4	87.3	83.8	85.5
DPTText-DETR	83.0	73.7	78.1	91.8	86.4	89.0	91.7	86.2	88.9
SwinTextSpotter	–	–	–	–	–	88.0	–	–	88.0
SwinTextSpotter v2	–	–	–	91.1	84.5	87.7	–	–	88.2
StarNet-Tex	–	–	–	–	–	86.5	–	–	85.1
CADet	83.3	76.0	79.5	90.9	88.0	89.4	92.2	86.4	89.2

Figure 6 further shows the type of instances represented by the ArT improvement. The examples contain curved logos, dense product text, low-contrast signs, and background structures that visually resemble text. The detected regions remain close to character strokes rather than expanding to the surrounding object boundary. This observation matches the quantitative result in Table 2: the gain over DPTText-DETR is modest in F-measure, but it appears in challenging arbitrary-shaped cases where semantic guidance helps select plausible text boundaries.

**Figure 6.** Detection examples on ArT. The visualized results show that character-guided queries preserve curved and low-contrast text instances in cluttered scenes.

4.3. Ablation Analysis for Recognition

The recognition ablations evaluated the permutation number in PLM, the mask strategy, and the effect of CLIE. Six permutations produced the most stable overall result, particularly on irregular datasets. As shown in Table 3, CLIE improved accuracy on most benchmarks, with especially clear gains on SVTP and CUTE. CUTE accuracy increased from 96.2% without CLIE to 99.0% with CLIE, showing that serialized local image information was useful for curved or irregular text.

The strongest recognition gains occurred where global word-shape cues were less reliable. CLIE improved SVTP by 1.4 percentage points and CUTE by 2.8 percentage points. It also improved both ICDAR2015 evaluation sets, which contain low-resolution and blurred text. These results support the design choice of inserting local visual evidence into character-position queries rather than relying only on global image tokens.

Table 3. Effect of CLIE on regular and irregular recognition benchmarks.

CLIE	IIIT5K	SVT	IC13_857	IC13_1015	IC15_1811	IC15_2077	SVTP	CUTE
No	99.0	98.5	98.1	98.1	90.6	89.5	94.9	96.2
Yes	98.9	98.6	99.0	99.0	91.1	89.9	96.3	99.0

4.4. Comparison with Scene Text Recognizers

Table 4 reports recognition accuracy on regular and irregular datasets. SIETR achieved the best or tied-best result on SVT, IC13_857, IC13_1015, IC15_1811, IC15_2077, SVTP, and CUTE among the methods with reported values. Compared with PARSeq, SIETR improved SVTP from 95.2% to 96.3%, CUTE from 98.6% to 99.0%, IC15_1811 from 90.3% to 91.1%, and IC15_2077 from 89.1% to 89.9%. Compared with CDistNet, SIETR improved SVTP and CUTE by 2.8 and 2.1 percentage points, respectively; compared with SVTRv2-B, SIETR improved SVTP by 2.8 percentage points. Unlike character-distance modeling in CDistNet or CTC-oriented feature alignment in SVTRv2-B, SIETR fuses serialized local image embeddings with semantic queries, giving the decoder stronger character-level visual grounding on irregular text. On IIIT5K, SIETR was slightly lower than PARSeq by 0.2 percentage points. This result suggests that its advantage is more pronounced in irregular and distorted scenes than in cleaner regular text.

Table 5 demonstrates that SIETR obtains competitive recognition accuracy with only 3.2 G FLOPs, which is lower than PARSeq with the same number of parameters. This computational efficiency of SIETR mainly benefits from the compact visual-context interaction strategy. Although CLIE introduces additional lightweight operations, including linear projection and Top-K local feature selection, it reduces redundant interactions between character queries and dense visual tokens during decoding. Therefore, the joint decoder operates on character-related local embeddings rather than repeatedly processing the full visual representation, leading to lower overall computational complexity.

Table 4. Recognition comparison on regular and irregular text benchmarks.

Model	IIIT5K	SVT	IC13_857	IC13_1015	IC15_1811	IC15_2077	SVTP	CUTE
ViTSTR-S	98.1	95.8	97.6	97.7	88.4	87.1	91.4	96.1
CRNN	94.6	90.7	94.1	94.5	82.0	78.5	80.6	89.1
TRBA	98.6	97.0	97.6	97.6	89.8	88.7	93.7	97.7
ABINet	98.6	97.8	98.0	97.8	90.2	88.5	93.9	97.7
PARSeq	99.1	98.3	98.3	98.3	90.3	89.1	95.2	98.6
CDistNet	98.7	97.1	97.8	—	89.6	—	93.5	96.9
SVTRv2-B	99.2	98.0	98.7	—	91.1	—	93.5	99.0
SIETR	98.9	98.6	99.0	99.0	91.1	89.9	96.3	99.0

Table 5. Efficiency comparison for text recognition models. Average accuracy is computed as a sample-size-weighted average over the eight recognition benchmarks reported in Table 4.

Model	Average Accuracy (%)	Parameters (M)	FLOPs (G)
ViTSTR-S	93.5	85.8	17.6
TRBA	94.5	49.8	10.9
PARSeq	95.1	23.8	6.3
ABINet	94.6	36.9	7.3
SIETR	95.6	23.8	3.2

The decoder visualization in Figure 7 explains why the recognition gain is larger on SVTP and CUTE than on regular text datasets. The highlighted regions move along the character sequence and remain near the corresponding character strokes, even when the input word is blurred or curved. This behavior is consistent with the CLIE ablation in Table 3: adding local image embeddings improves SVTP by 1.4 percentage points and CUTE by 2.8 percentage points, while producing only a negligible change on IIIT5K. The qualitative and quantitative evidence therefore points to the same mechanism, namely that local visual grounding is most valuable when global word shape and language context are insufficient.

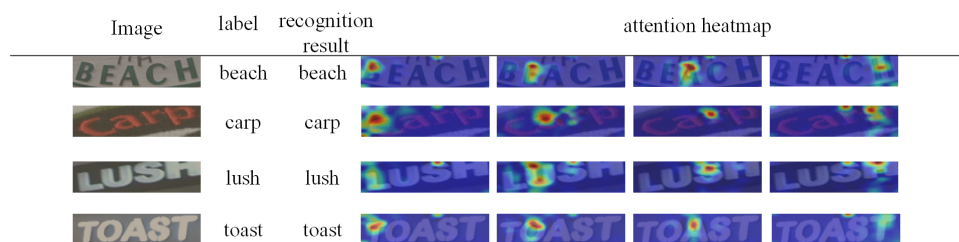


Figure 7. Attention visualization from the joint decoder. The highlighted regions indicate that serialized local image embeddings keep character predictions aligned with corresponding visual evidence.

5. Discussion

The results demonstrate that explicit information improves detection and recognition when introduced before the final prediction stage. In CADet, character-level semantics guide boundary refinement, improving recall on challenging text instances. On ArT, the CIA module increases recall from 73.4% to 75.6%, and the full model achieves a 79.5% F-measure. The improvements are mainly attributed to enhanced text-region responses and character-guided queries that preserve curved and weak text boundaries, rather than increased model size.

For recognition, SIETR shows larger gains on irregular benchmarks, where global context is less reliable. It improves SVTP, CUTE, and ICDAR2015 over PARSeq with fewer FLOPs and the same parameter scale. The decoder visualization confirms that serialized local image embeddings improve character-level visual grounding.

Overall, detection benefits from character semantics for separating text from background, while recognition benefits from local visual guidance for reliable character prediction. This asymmetric explicit-guidance design provides a unified strategy for robust scene text analysis.

6. Conclusions

This paper presents explicit information mechanisms for scene text analysis. CADet introduces character-level semantic guidance into arbitrary-shaped text detection via text-enhanced features, adaptive character guidance, and position/classification compensation. SIETR introduces serialized local image embeddings into STR through permutation-based semantic modeling and character-local visual alignment. Across public benchmarks, CADet improved the F-measure on ArT, Total-Text, and CTW1500, achieving 79.5%, 89.4%, and 89.2%, respectively, while SIETR achieved 95.6% sample-size-weighted average recognition accuracy with 23.8 M parameters and 3.2 G FLOPs. The results support explicit character-level visual–context alignment as a compact strategy for curved, distorted, and cluttered scene text.

Author Contributions: Conceptualization, L.C. and D.C.; methodology, H.L.; formal analysis, L.C. and H.L.; writing—original draft preparation, L.C. and H.L.; writing—review and editing, D.C.; supervision, D.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of China under Grants 62571134, and by the Fuzhou Industry-Department Project under Grant 2025-ZD-020.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

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

References

- Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
- Cheng, Y.; Huang, J.; Tai, I.S.; Mondal, S.K.; Wang, T.; Kabir, H.M.D. A two-stage end-to-end framework for robust scene text spotting with self-calibrated detection and contextual recognition. *Electronics* **2025**, *14*, 4594. [\[CrossRef\]](#)
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Neural Information Processing Systems Foundation, Inc.: San Diego, CA, USA, 2017; Volume 30.
- 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, Las Vegas, NV, USA, 27 June–1 July 2016; pp. 770–778.
- Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*; International Conference on Learning Representations Foundation (ICLRF): San Francisco, CA, USA, 2021.
- Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*; International Conference on Learning Representations Foundation (ICLRF): San Francisco, CA, USA, 2021.
- Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
- 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, Munich, Germany, 8–14 September 2018; pp. 20–36.
- 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 International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8440–8449.
- Kobchaisawat, T.; Chalidabhongse, T.H.; Satoh, S. Scene text detection with polygon offsetting and border augmentation. *Electronics* **2020**, *9*, 117. [\[CrossRef\]](#)
- 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, Long Beach, CA, USA, 16–20 June 2019; pp. 9365–9374.
- 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, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11474–11481.
- Liu, Y.; Chen, H.; Shen, C.; He, T.; Jin, L.; Wang, L. ABCNet: Real-time scene text spotting with adaptive Bezier-curve network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 9809–9818.
- Tang, H.; Li, Z.; Zhang, D.; He, S.; Tang, J. Divide-and-Conquer: Confluent Triple-Flow Network for RGB-T Salient Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, *47*, 1958–1974. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhang, X.; Su, Y.; Tripathi, S.; Tu, Z. Text spotting transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 9519–9528.
- 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 Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 3241–3249.
- Huang, M.; Liu, Y.; Peng, Z.; Liu, C.; Lin, D.; Zhu, S.; Yuan, N.; Ding, K.; Jin, L. SwinTextSpotter: Scene text spotting via better synergy between text detection and text recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 4593–4603.
- Huang, M.; Peng, D.; Li, H.; Peng, Z.; Liu, C.; Lin, D.; Liu, Y.; Bai, X.; Jin, L. SwinTextSpotter v2: Towards better synergy for scene text spotting. *Int. J. Comput. Vis.* **2025**, *133*, 5281–5301. [\[CrossRef\]](#)
- Hu, Z.; Zhao, Y.; Yang, Z.; Jiang, J.; Xu, X. StarNet-Tex: Lightweight real-time text detection via middle-order enhanced contextual attention. *Complex Intell. Syst.* **2025**, *11*, 432. [\[CrossRef\]](#)
- Cai, Y.; Zhou, F.; Yin, R. Exploring style-robust scene text detection via style-aware learning. *Electronics* **2024**, *13*, 243. [\[CrossRef\]](#)
- Shi, B.; Bai, X.; Yao, C. 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.* **2017**, *39*, 2298–2304. [\[CrossRef\]](#) [\[PubMed\]](#)
- Atienza, R. Vision Transformer for fast and efficient scene text recognition. In Proceedings of the International Conference on Document Analysis and Recognition, Lausanne, Switzerland, 5–10 September 2021; pp. 319–334.
- Fang, S.; Xie, H.; Wang, Y.; Mao, Z.; Zhang, Y. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 July 2021; pp. 7098–7107.

24. Bautista, D.; Atienza, R. Scene text recognition with permuted autoregressive sequence models. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; pp. 178–196.
25. Feng, Y.; Li, C. Towards robust scene text recognition: A dual correction mechanism with deformable alignment. *Electronics* **2025**, *14*, 3968. [[CrossRef](#)]
26. Peng, H.; Yu, J.; Nie, Y. Efficient neural network for text recognition in natural scenes based on end-to-end multi-scale attention mechanism. *Electronics* **2023**, *12*, 1395. [[CrossRef](#)]
27. Zheng, T.; Chen, Z.; Fang, S.; Xie, H.; Jiang, Y.-G. CDistNet: Perceiving multi-domain character distance for robust text recognition. *Int. J. Comput. Vis.* **2024**, *132*, 300–318. [[CrossRef](#)]
28. Du, Y.; Chen, Z.; Xie, H.; Jia, C.; Jiang, Y.-G. SVTRv2: CTC beats encoder-decoder models in scene text recognition. In Proceedings of the 2025 IEEE/CVF International Conference on Computer Vision (ICCV), Honolulu, HI, USA, 21–25 October 2025; pp. 20147–20156. [[CrossRef](#)]
29. Ch'ng, C.K.; Chan, C.S.; Liu, C.-L. Total-Text: Toward orientation robustness in scene text detection. *Int. J. Doc. Anal. Recognit.* **2020**, *23*, 31–52. [[CrossRef](#)]
30. Liu, Y.; Jin, L.; Zhang, S.; Zhang, S.; Luo, C. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognit.* **2019**, *90*, 337–345. [[CrossRef](#)]
31. Ch'ng, C.K.; Liu, Y.; Sun, Y.; Ng, C.C.; Luo, C.; Ni, Z.; Fang, C.; Zhang, S.; Han, J.; Ding, E.; et al. ICDAR2019 robust reading challenge on arbitrary-shaped text. In Proceedings of the International Conference on Document Analysis and Recognition, Sydney, Australia, 20–23 September 2019; pp. 1571–1576.
32. Mishra, A.; Alahari, K.; Jawahar, C.V. Scene text recognition using higher order language priors. In Proceedings of the British Machine Vision Conference, Surrey, UK, 3–7 September 2012.
33. Wang, K.; Babenko, B.; Belongie, S. End-to-end scene text recognition. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 1457–1464.
34. Phan, T.Q.; Shivakumara, P.; Tian, S.; Tan, C.L. Recognizing text with perspective distortion in natural scenes. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 569–576.
35. Risnumawan, A.; Shivakumara, P.; Chan, C.S.; Tan, C.L. A robust arbitrary text detection system for natural scene images. *Expert Syst. Appl.* **2014**, *41*, 8027–8048. [[CrossRef](#)]
36. Karatzas, D.; Gomez-Bigorda, L.; Nicolaou, A.; Ghosh, S.; Bagdanov, A.; Iwamura, M.; Matas, J.; Neumann, L.; Chandrasekhar, V.R.; Lu, S.; et al. ICDAR 2015 competition on robust reading. In Proceedings of the International Conference on Document Analysis and Recognition, Nancy, France, 23–26 August 2015; pp. 1156–1160.

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.