https://iopscience.iop.org/article/10.1088/1402-4896/ae94bb/meta

PAPER

Multi-query text spotter transformer for arbitrary shape scene text detection and recognition

Canhui Xu*, Yaqi Chen, Jialiang Li and Cao Shi

Published 12 August 2026 • © 2026 IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

Physica Scripta, Volume 101, Number 32

Citation Canhui Xu *et al* 2026 *Phys. Scr.* **101** 325002

DOI 10.1088/1402-4896/ae94bb

Could you publish open access in this journal at no cost?

See if your institution is one of many in Hong Kong covering the cost of APCs in this journal.

Authors References Open science

Article information

Abstract

References

Article metrics

12 Total downloads

Submit

Submit to this Journal

Permissions

Get permission to re-use this article

Share this article

Abstract

Control points regression based detection task, which predicts Bezier curve or polygon control points to locate text position, was considered preliminary step for scene text recognition. Recent Transformer based methods jointly connect the detection and recognition within one unified network. To explore the task relational contexts and describe the mutual effects, in this work, we propose a novel Multi-Query TExt Spotter TRansformer (Multi-Query TESTR), which consists of a shared transformer encoder and a single Multi-Query Transformer Decoder. Both detection and recognition tasks shared the same transformer encoder to process the multi-scale inputs with positional embedding. Detection queries focus on spatial awareness across multiple feature scales to regress control points. Conversely, recognition queries capture the semantic features of text instances. To learn the mutual effects between these task related queries, the cross task attention is obtained by self attention transformer for reasoning dependencies among these tasks. With the fused interaction of multiple queries, two

You may also like

JOURNAL ARTICLES

Learning Deeply Supervised Scene Text Detectors from Scratch

Fast Arbitrary Shaped Scene Text Detection via Text Discriminator

A Parallel Backbone Networks Structure for Scene Text Detection

A scene text detection method based on multi-scale context enhancement and boundary guidance

An Attention-based Text Detection and Recognition Method for Terminals of Current Transformer's Secondary Circuit

physicsworld | jobs

Faculty Positions at Institute of Physics (IOP), Chinese Academy of Sciences
Institute of Physics, Chinese Academy of Sciences

Postdoctoral Researcher positions – Statistical Physics and Active Matter –
Max Planck Institute for Dynamics and Self-Organization

Scientific Director
Istituto Italiano di Tecnologia

More jobs Post a job

Journal Name

Crossmark

RECEIVED
dd Month yyyy

REVISED
dd Month yyyy

RESEARCH ARTICLE

Multi-Query Text Spotter Transformer for Arbitrary Shape Scene Text Detection and Recognition

Canhui Xu[1,*], Yaqi Chen[1], Jialiang Li[1] and Cao Shi[1]

[1]Qingdao University of Science and Technology, Laoshan Campus, Qingdao 266100, China
[*]Corresponding author:Canhui Xu

E-mail: ccxu09@yeah.net

Keywords: Scene Text Detection and Recognition, Arbitrarily Shaped and Multi-Oriented Scene Text, Multi-task Prediction, Multi-Query Learning.

Abstract

Control points regression based detection task, which predicts Bezier curve or polygon control points to locate text position, was considered preliminary step for scene text recognition. Recent Transformer based methods jointly connect the detection and recognition within one unified network. To explore the task relational contexts and describe the mutual effects, in this work, we propose a novel Multi-Query TExt Spotter TRansformer (Multi-Query TESTR), which consists of a shared Transformer Encoder and a single Multi-Query Transformer Decoder. Both detection and recognition tasks shared the same transformer encoder to process the multi-scale inputs with positional embedding. Within the decoder, the learning of task-specific queries is decoupled. Detection queries focus on spatial awareness across multiple feature scales to regress control points. Conversely, recognition queries capture the semantic features of text instances. To learn the mutual effects between these task related queries, the cross task attention is obtained by self attention transformer for reasoning dependencies among these tasks. With the fused interaction of multiple queries, two separate branches are designed to accomplish the specific detection and recognition query learning. After concatenation of the two branches, Collaborative Multi-Scale Deformable Cross Attention is performed to learn the cross attention between queries and feature contents. Experiments conducted on three publicly available benchmarks SCUT-CTW1500, Total-Text, and ICDAR15 have demonstrated that Multi Query TESTR achieves state-of-the-art performance on arbitrary shaped and multi-oriented scene text detection and recognition. Our code will be released on Github.

1 Introduction

Scene text detection and recognition are two fundamental tasks in automatic scene text reading, which profits various real-world multimedia applications. For instance, [1] provide a comprehensive review of the deep learning era in this field, while [2] and [3] offer detailed surveys on detection and recognition algorithms and their practical implications in imagery. Previous traditional learning methods treat the two tasks as successive processes, considering text detection as a preliminary step for scene text recognition, where features are learned separately for each task. However, humans are naturally good at detecting and recognizing text simultaneously. Achieving multiple tasks at the same time has attracted growing attention in computer vision. Specifically, single-point spotting frameworks like SPTS [4] and SPTS v2 [5] have been proposed to simplify the pipeline, while multi-task Transformer architectures such as TTS [6], TESTR [7], and DeepSolo [8] have been developed to unify detection and recognition within a single model. Multi-Task Dense Prediction (MTDP) further explores learning shared feature representations and task interactions, as discussed by [9].

Most early text spotters rely heavily on RoI (Region of Interests) strategies to learn candidate proposals. Mask TextSpotter [10] utilizes an instance-level detector head and another separate character-level recognizer head. It makes mask predictions inside an RoI to segment text instances and then features are further sampled and grouped for recognition. On the other hand, control points regression-based methods parameterize the text instances as a group of coordinates.

TextDragon [11] proposes RoISlide to unify the heads. ABCNet [12, 13] proposes BezierAlign, and the controlling points are represented by Bezier curve points.

Recent Transformer-based structures gain increasing popularity in the design of text-spotter pipelines. TTS [6] utilizes one decoder for both detection and recognition tasks but requires an additional segmentation head and RNN module for text recognition. SPTS [4, 5] proposes an end-to-end scene text spotting method simplifying the training model to learn a single-point for each text instance, and its token sequence contains both detection and recognition results. TESTR [7] is proposed to employ dual decoders for detection and recognition separately, while the backbone and transformer encoder parts share parameters in both detection and recognition tasks. However, modeling multiple decoders demands parallel computation complexity, which is more complex and computationally expensive. DeepSolo [8] and ESTextSpotter [14] represent recent attempts to use a single decoder for both detection and recognition tasks by shared queries or task-aware queries. They depict tasks interaction relation implicitly or explicitly with shared queries. Considering the synergy and distinct characteristics of text detection and recognition, we employ multi-query strategy to deal with the relation of two sub-tasks.

To make the Transformer adaptable to the synergistic requirements of detection and recognition, we propose the Multi-Query TExt Spotter TRansformer (Multi-Query TESTR). Our framework consists of a Shared Transformer Encoder and a single Multi-Query Transformer Decoder. Unlike previous decoupled architectures, we introduce a Concate-Separate-Collaborative (CSC) learning paradigm within the decoder.

While these multi-task Transformer architectures have significantly advanced the field, they exhibit inherent architectural limitations. For instance, TESTR employs parallel dual decoders, which process detection and recognition independently, thereby lacking deep, early-stage dynamic interactions[7]. Conversely, DeepSolo utilizes a single decoder with highly entangled shared queries[8]. Although this explicitly couples the two tasks, it may restrict the representational flexibility required to capture the distinct, task-aware characteristics of localization and semantic recognition. To address these fundamental limitations and make the Transformer adaptable to the synergistic requirements of detection and recognition, we propose the Multi-Query TExt Spotter TRansformer (Multi-Query TESTR).

Specifically, the CSC module first concatenates task-specific queries to establish a unified feature space, then separates them for specialized learning, and finally facilitates deep collaboration through a Multi-task Graph Learning (MTGL) mechanism. It is worth noting that while graph-based feature aggregation has been explored in general dense prediction tasks [15], our MTGL is specifically tailored for the unique challenges of text spotting. It extends beyond homogeneous feature passing by explicitly modeling the highly heterogeneous dependencies between spatial detection queries and semantic recognition queries. Subsequently, Collaborative Multi-scale Deformable Cross-attention is performed to fuse query-to-content information effectively. Experiments conducted on three benchmark datasets demonstrate that Multi-Query TESTR achieves state-of-the-art performance while maintaining high computational efficiency.

The main contributions of this work are summarized as follows:

- We propose a novel CSC module within the decoder that facilitates unified task processing. It systematically integrates Multi-Task Query-to-Query Learning, Task-Specific Query Learning, and Collaborative Content-to-Query Learning, allowing the model to capture deep interactions between detection and recognition features.
- We introduce a graph-based mechanism that dynamically models the dependencies between task-specific queries. Unlike prior works that treat tasks independently, MTGL fosters collaborative optimization, enabling a synergistic effect where the two tasks mutually enhance each other.
- We design a streamlined Multi-Query Transformer Decoder that jointly handles detection and recognition. By eliminating the need for task-specific decoders (e.g., the dual-decoder design in TESTR), our architecture significantly reduces structural redundancy while maintaining state-of-the-art performance.

2 Related Work

Detection and recognition tasks are usually trained separately and joined as a successive procedure. An end-to-end unified framework is more desired to accomplish the two tasks simultaneously. Increasingly, research efforts are directed towards jointly considering text detection and recognition, referred to as end-to-end text detection and recognition or text spotting.

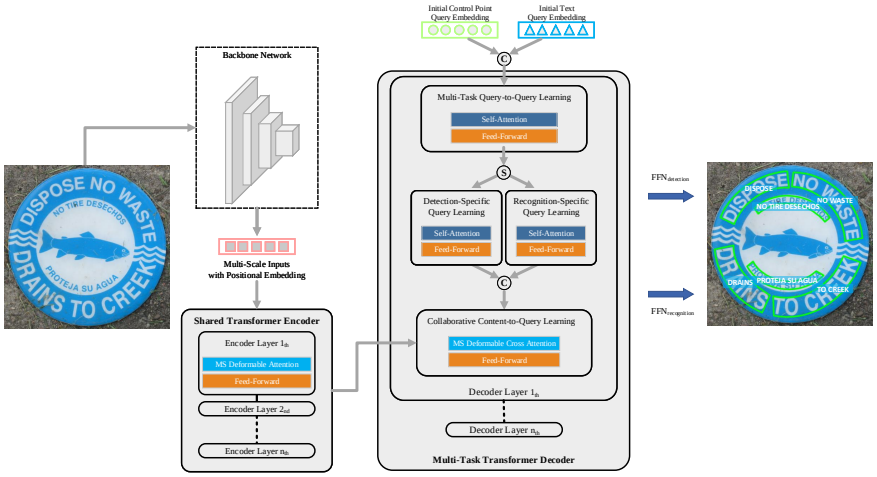

Figure 1: The overall architecture of Multi-Query TESTR, which consists of a shared Transformer Encoder and a single Multi-Query Transformer Decoder.

The Mask TextSpotter series [10, 16] utilize character-level supervision for detection and recognition. The ABCNet series [12, 13] use BezierAlign to handle curved text, which requires the prediction of a small fixed number of points. However, they require additional RoI-based or TPS-based connectors, and the only shared part between the detector and recognizer is the backbone network features. HGR-Net[17, 18] explores hierarchical text instance-level and component-level representation for arbitrarily-shaped scene text detection.

With the impressive success of Transformers in visual tasks, more recent works adopt Transformer-based structures for the Text Spotting task. DEtection TRansformer (DETR) [19] removed the proposal anchors and post-processing Region-Of-Interest procedures, which made a profound contribution to the vision transformers. TExt Spotter TRansformer(TESTR) [7] proposed a unified vision transformer for scene text detection and character recognition, which consists of a single encoder and two parallel decoders. The shared encoder extracts feature representation from multiple scales, which is fed into dual decoders for predicting the control points and characters. TTS [6] utilizes an encoder and a decoder with multiple prediction heads (RNN, FFN, DeConv) for performing various tasks such as detection and recognition, and replaces NMS with a bipartite graph matching algorithm for post-processing. SPTS [4] and its improved version SPTSv2 [5] model end-to-end text detection and recognition tasks as a concise sequence prediction problem and integrate text detection and recognition into a Transformer-based sequence prediction model without using bipartite graph matching.

DeepSolo[8] and ESTextSpotter[14] represent recent attempts to use a single decoder for both detection and recognition tasks by shared queries or task-aware queries. ESTextSpotter categorizes these methods as implicit synergy methods because they use shared parameters for detection and recognition tasks during the initialization of decoder queries. However, heavily shared queries often struggle to balance the distinct representational needs of precise geometric regression and abstract semantic classification. In contrast to both the decoupled dual-decoder design of TESTR[7] and the rigidly shared query design of DeepSolo[8], our proposed Multi-Query TESTR introduces a structured Concate-Separate-Collaborative (CSC) paradigm. By dynamically concatenating, explicitly separating, and graph-collaborating the queries, our method ensures that both the synergy and task-aware distinct characteristics are preserved and optimally leveraged for multi-task dense prediction. Furthermore, while multi-query graph learning has recently been explored to handle task relationships in generic dense prediction (e.g., semantic segmentation) [15], adapting such mechanisms to scene text spotting presents unique challenges due to the extreme heterogeneity of the sub-tasks (i.e., spatial coordinate regression versus semantic character sequence recognition). To effectively bridge this semantic-spatial gap, our architecture extends beyond standard homogeneous feature aggregation by introducing a novel Mutual Gradient Gating mechanism, ensuring noise-resilient cross-task optimization tailored specifically for text spotting.

3 Method

The architecture of our Multi-Query TESTR is presented in Figure 1. Specifically, for each input image, backbone network such as ResNet, Swin-T, ViTAE, is utilized to extract multi-scale

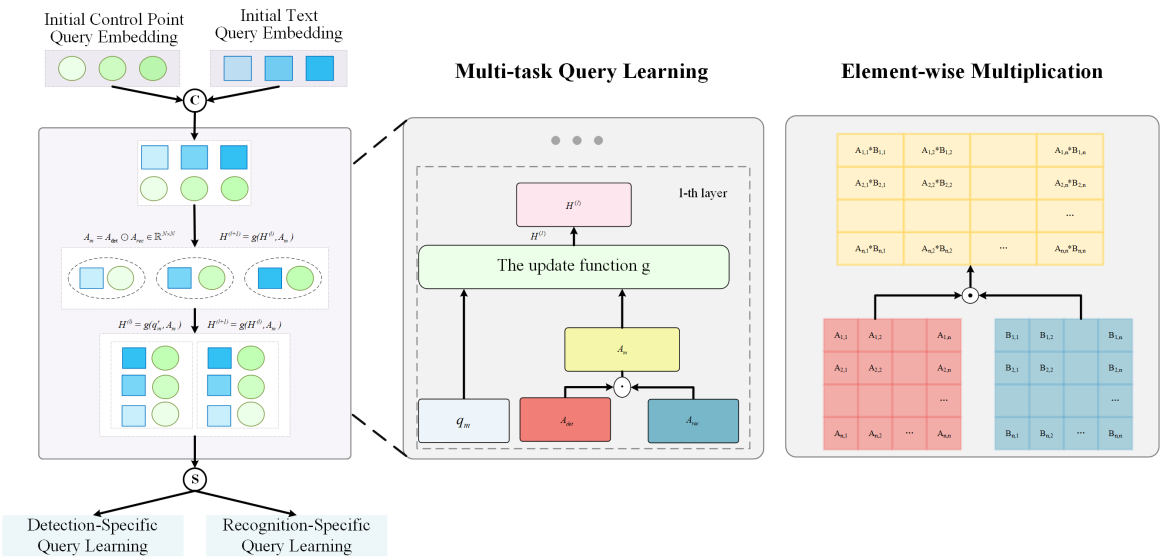

Figure 2: Detailed illustration of the Multi-Task Query-to-Query Learning module. **Left:** The overall macroscopic pipeline demonstrating the Concate-Separate-Collaborative (CSC) process, where initial control point and text queries are merged, updated, and separated. **Right:** A zoomed-in view of the internal microscopic mechanisms within the multi-task graph learning layer, explicitly highlighting the layer-wise node update function g and the element-wise multiplication strategy for fusing task-specific adjacency matrices ($\mathbf{A}_m = \mathbf{A}_{det} \odot \mathbf{A}_{rec}$).

deformable feature maps, and shared Transformer encoder learns feature embeddings with contextual information from element contents. Then, a single Multi-Query Transformer decoder explores both the synergy and task-aware distinct characteristics for detection and recognition problems. Multi-Task Query-to-Query Learning is then separated into two parallel sub-tasks: Detection-Specific Query Learning and Recognition-Specific Query Learning. The concatenated Collaborative Content-to-Query Learning is performed to learn the cross attention between queries and feature contents.

3.1 Transformer Encoder

Denote the extracted feature maps as $\mathbf{X} = \left\{\mathbf{x}_l \in \mathbb{R}^{c_l \times h_l \times w_l} \mid l \in \{1, \ldots, L\}\right\}$, where l indexes the feature level, and set channel dimension c_l to d for each feature map. We reshape each feature map $\mathbf{x}_l$ into a sequence of flattened feature embeddings and transpose them described as $\left\{\hat{\mathbf{x}}_l \in \mathbb{R}^{h_l w_l \times d} \mid l \in \{1, \ldots, L\}\right\}$.

To exploit multi-scale feature representation, Transformer encoders integrate Multi-Scale Deformable Attention Modules presented in Deformable DETR [19] and Feed-Forward Networks (FFN). The MSDM-Attn can be illustrated as:

$$\begin{aligned} &\text{MSDM-Attn}(\mathbf{q}_{det}, \mathbf{p}_{q_{det}}, \{\hat{\mathbf{x}}_l\}_{l=1}^{L}) \\ &= \sum_{h=1}^{H} \mathbf{W}_h \left[\sum_{l=1}^{L} \sum_{k=1}^{K} A_{hlk}(\mathbf{q}_{det}) \cdot \mathbf{W}'_h \hat{\mathbf{x}}_l(\Phi_l(\mathbf{p}_{q_{det}}) + \Delta\mathbf{p}_{hlk}(\mathbf{q}_{det})) \right], \end{aligned} \tag{1}$$

where $\mathbf{p}_{q_{det}}$ is the normalized coordinates of the reference point for the query $\mathbf{q}_{det}$. And h, l, k are indices for the attention head, input feature level, and sampling point respectively. A_{hlk} denotes the attention weight for query $\mathbf{q}_{det}$, normalized with respect to K sampling points. Φ_l maps the normalized coordinates to the scale of the l-th level feature map, and $\Delta\mathbf{p}$ generates an appropriate sampling offset for the query. Both of them are added to form the sampling location for the feature map. $\mathbf{W}_h$ and $\mathbf{W}'_h$ are trainable weight matrices that are similar to those present in the original multi-head attention.

The formulation of each encoder layer can be expressed as:

$$\begin{aligned} &\mathbf{X}_e = \text{FFN}\left(\text{MSDM-Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V})\right), \\ &\mathbf{Q} = \mathbf{q}_{det}, \quad \mathbf{K} = \mathbf{p}_{q_{det}}, \quad \mathbf{V} = \{\hat{\mathbf{x}}_l\}_{l=1}^{L}. \end{aligned} \tag{2}$$

3.2 Multi-Query Decoder

Following TESTR [7], we first generate initialized control point query embeddings and text query embedding representations for scene text detection and recognition. Subsequently, the initialized detection queries and recognition queries are sent into the decoder as input, and outputs the decoded feature representations for detecting and recognizing text instances. Each decoder layer integrates Multi-Task Query-to-Query Learning, Detection-Specific Query Learning, Recognition-Specific Query Learning, and Collaborative Content-to-Query Learning.

3.2.1 Multi-task query learning. Multi-task query learning consists of a multi-task graph learning (MTGL) layer, a self-attention mechanism, and a feed-forward network. To improve readability and systematically describe the query interaction, we detail the MTGL process in three structured steps: Task-Specific Graph Construction, Multi-Task Adjacency Matrix Fusion, and Graph Convolution and Query Update.

Task-Specific Graph Construction. The graph construction module builds the detection task graph $\mathbf{G}_{det} = (\mathbf{V}_{det}, \mathbf{E}_{det})$ using candidate reference coordinate points, and constructs the recognition task graph $\mathbf{G}_{rec} = (\mathbf{V}_{rec}, \mathbf{E}_{rec})$ using the recognition task query embedding representation $\mathbf{q}_{rec}$.

For N candidate targets, their reference points generated after encoding can be represented as bounding boxes $x = \{(x_i, y_i, w_i, h_i)\}_{i=1}^{N}$. By constructing $\mathbf{G}_{det}$ to capture the spatial relationship information implied in the geometric attributes of scene text, the center coordinates $\{(x_c^i, y_c^i)\}_{i=1}^{N}$ are calculated as:

$$(x_c^i, y_c^i) = \left(x_i + \frac{w_i}{2}, y_i + \frac{h_i}{2}\right) \tag{3}$$

Then, the spatial correlation R_{det} is encoded by measuring the Euclidean distance between graph nodes $(\mathbf{v}_{det}^i, \mathbf{v}_{det}^j)$, calculated as:

$$R_{det} = \sqrt{(\Delta x_c^{ij})^2 + (\Delta y_c^{ij})^2} \tag{4}$$

$$\Delta x_c^{ij} = \frac{x_c^i}{w_i} - \frac{x_c^j}{w_j}, \quad \Delta y_c^{ij} = \frac{y_c^i}{h_i} - \frac{y_c^j}{h_j} \tag{5}$$

Next, the spatial correlation R_{det} is normalized to the interval $(0, 1)$, so the adjacency matrix of the detection task graph $\mathbf{G}_{det}$ can be represented as $\mathbf{A}_{det} = \mathrm{Norm}(R_{det}) \in \mathbb{R}^{N \times N}$.

Simultaneously, to model contextual information between texts, the initialized text query is projected into a latent feature space to encode the semantic relationships at the recognition task level $\mathbf{E}_{rec}$:

$$\mathbf{o}_i = \theta(\mathbf{q}_{rec}^i), \quad i \in \{1, \ldots, N\} \tag{6}$$

where $\mathbf{q}_{rec}^i$ is the text query embedding representation after dimension transformation and reshaping, and θ is the non-linear transformation that projects $\mathbf{q}_{rec}^i$ into the latent feature space. Let $\mathbf{O} = \{\mathbf{o}_i\}_{i=1}^{N} \in \mathbb{R}^{N \times C_{rec}}$ be the set of these projected elements, where C_{rec} is the number of channels in the latent vectors. After matrix multiplication, the semantic relationship $\mathbf{E}_{rec}$ is calculated as $\mathbf{E}_{rec} = \mathbf{O}\mathbf{O}^T$. Therefore, the adjacency matrix of the recognition task graph $\mathbf{G}_{rec}$ can be represented as $\mathbf{A}_{rec} = \mathbf{E}_{rec} \in \mathbb{R}^{N \times N}$.

Multi-Task Adjacency Matrix Fusion. To capture the synergistic relationship information, we fuse the adjacency matrices of the two specific tasks using element-wise multiplication ($\odot$) to generate the multi-task adjacency matrix $\mathbf{A}_m$:

$$\mathbf{A}_m = \mathbf{A}_{det} \odot \mathbf{A}_{rec} \in \mathbb{R}^{N \times N} \tag{7}$$

The choice of element-wise multiplication ($\odot$) for adjacency fusion is further justified from the perspective of gradient propagation. According to the chain rule in backpropagation, the partial derivatives of the fused adjacency $\mathbf{A}_m$ with respect to each task-specific adjacency are given by:

$$\frac{\partial \mathbf{A}_m}{\partial \mathbf{A}_{det}} = \mathbf{A}_{rec}, \quad \frac{\partial \mathbf{A}_m}{\partial \mathbf{A}_{rec}} = \mathbf{A}_{det} \tag{8}$$

This formulation establishes a Mutual Gradient Gating mechanism, where the gradient flow of one task is dynamically scaled by the forward activation of the other. Specifically, when the recognition branch ($\mathbf{A}_{rec}$) identifies a high-confidence semantic correlation, it naturally amplifies the learning signal for the corresponding spatial nodes in the detection branch ($\mathbf{A}_{det}$). Conversely, if either task identifies a node relationship as noise (where activation ≈ 0), the multiplication effectively

suppresses the gradient backpropagation for that connection. Compared to additive fusion, this gated interaction leads to a more synergistic and noise-resilient optimization landscape.

Graph Convolution and Query Update. Using channel-level concatenation, we integrate the initialized detection task query $\mathbf{q}_{det}$ and recognition task query $\mathbf{q}_{rec}$ to generate the overall query embedding $\mathbf{q}_m$ for multi-task query-to-query learning. The multi-task graph learning layer first performs dimension transformation on $\mathbf{q}_m$, then aggregates the multi-task aware relationship features using the transformed query embedding $\mathbf{q}_m^r$ and the multi-task adjacency matrix $\mathbf{A}_m$.

Specifically, the graph convolution neural network node update function g is applied layer by layer:

$$\mathbf{H}^{(l)} = g(\mathbf{q}_m^r, \mathbf{A}_m), \quad \mathbf{H}^{(l+1)} = g(\mathbf{H}^{(l)}, \mathbf{A}_m) \tag{9}$$

where $\mathbf{H}^{(l)}$ is the input and $\mathbf{H}^{(l+1)}$ is the output of the l-th layer. After applying the graph convolution operation, the specific update function g is calculated as:

$$\mathbf{H}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}) \tag{10}$$

where σ denotes the LeakyReLU activation function, $\mathbf{W}^{(l)}$ is the learned weight transformation matrix, $\tilde{\mathbf{A}} = \mathbf{A}_m + \mathbf{I}_N$ is the multi-task adjacency matrix with added self-connections, and $\tilde{\mathbf{D}} \in \mathbb{R}^{N\times N}$ is the corresponding degree matrix.

After multi-task graph learning, we perform dimension transformation again on the generated graph embedding representation $\mathbf{H}^{(l+1)}$, denoted as $r(\mathbf{H}^{(l+1)})$, and then pass it into the self-attention mechanism and feed-forward network layers to generate the final updated multi-task feature representation $\mathbf{X}_m$:

$$\mathbf{X}_m = \text{FFN}(\text{Self-Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V})) \tag{11}$$

$$\mathbf{Q} = \mathbf{K} = \mathbf{V} = r(\mathbf{H}^{(l+1)}) \tag{12}$$

Remark 1. The core innovation of our Multi-Query decoder lies in the Concate-Separate-Collaborative (CSC) learning paradigm. Unlike existing methods (e.g., TESTR) that utilize decoupled dual decoders for detection and recognition, our CSC module facilitates a unified feature space where task-specific queries are first concatenated for global interaction and then separated for specialized refinement. This allows the subsequent Multi-task Graph Learning (MTGL) mechanism to dynamically model the inherent dependencies between tasks, enabling a synergistic effect that promotes mutual optimization within a single decoder.

3.2.2 Detection and Recognition task query Learning. We split the learned multi-task feature embeddings $\mathbf{X}_m$ into two sub-task queries $\mathbf{q}_{det}$ and $\mathbf{q}_{rec}$:

$$\{\mathbf{q}_{det}, \mathbf{q}_{rec}\} = \text{Split}(\mathbf{X}_m). \tag{13}$$

The task-specific query learning process can be described as:

$$\begin{aligned} \mathbf{X}_{det} &= \text{Self-Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}), \\ \mathbf{Q} = \mathbf{K} &= \mathbf{q}_{det} + \mathbf{pos}_{det}, \quad \mathbf{V} = \mathbf{q}_{det}, \\ \mathbf{X}_{rec} &= \text{Self-Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}), \\ \mathbf{Q} = \mathbf{K} &= \mathbf{q}_{rec} + \mathbf{pos}_{rec}, \quad \mathbf{V} = \mathbf{q}_{rec}. \end{aligned} \tag{14}$$

where $\mathbf{pos}_{det}$ and $\mathbf{pos}_{rec}$ are the corresponding position encodings to $\mathbf{q}_{det}$ and $\mathbf{q}_{rec}$.

3.2.3 Collaborative content into query learning. To perform collaborative cross attention, MSDM-Attn takes the concatenated query embedding $\mathbf{q}'_m$ and output of transformer encoder $\mathbf{X}_e$:

$$\begin{aligned} \mathbf{X}_d &= \text{MSDM-Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}), \\ \mathbf{Q} = \mathbf{q}'_m, &\quad \mathbf{K} = \hat{\mathbf{p}}_d, \quad \mathbf{V} = \mathbf{X}_e. \end{aligned} \tag{15}$$

3.3 Multi-Task Supervised Learning

Receiving queries $\mathbf{q}_{det}$ and $\mathbf{q}_{rec}$ from the decoder for each image, we adopt simple prediction heads to solve sub-tasks. We use the Hungarian algorithm to get an optimal injective function that minimizes the matching cost from prediction result $[\hat{\mathbf{Y}}]$ to ground truth $[\mathbf{Y}]$:

$$\arg\min_{\varphi} \sum_{g=0}^{G-1} \mathcal{C}\left(\mathbf{Y}^{(g)}, \hat{\mathbf{Y}}^{(\varphi(g))}\right), \tag{16}$$

where G is the number of ground truth instances per image. The loss function consists of the three losses:

$$\mathcal{L}_{total} = \sum_{k} \left(\alpha_{cls} \mathcal{L}_{cls}^{(k)} + \alpha_{char} \mathcal{L}_{char}^{(k)} + \alpha_{coord} \mathcal{L}_{coord}^{(k)} \right). \tag{17}$$

where α_{cls}, α_{coord}, and α_{char} are hyperparameters used to balance the multi-task loss terms. Since the scales of classification error, coordinate offset, and character probability vary significantly, these coefficients are essential for stable joint training. Specifically, we empirically set $\alpha_{cls} = 2.0$, $\alpha_{coord} = 5.0$, and $\alpha_{char} = 1.0$ in our experiments, which aligns with the common settings in previous Transformer-based spotters.

For text instance classification, the holistic query $\mathbf{q}_m$ is sent to a linear projection for binary classification (text or non-text). During inference, we take the mean of N scores as the confidence score for each instance. The loss is defined as:

$$\begin{aligned} \mathcal{L}_{\text{cls}}^{(j)} = & -\mathbb{1}_{\{j \in \text{Im}(\sigma)\}} \alpha \left(1 - \hat{b}^{(j)}\right)^{\gamma} \log\left(\hat{b}^{(j)}\right) \\ & -\mathbb{1}_{\{j \notin \text{Im}(\sigma)\}} (1-\alpha) \left(\hat{b}^{(j)}\right)^{\gamma} \log\left(1 - \hat{b}^{(j)}\right). \end{aligned} \tag{18}$$

For control point loss, a 3-layer MLP head MLP_{coord} with detection query $\mathbf{q}_{det}$ is used to predict the coordinate offsets from reference points to ground truth points on the center curve. $L1$ distance loss is used for control point coordinate regression:

$$\mathcal{L}_{\text{coord}}^{(j)} = \mathbb{1}_{\{j \in \text{Im}(\sigma)\}} \sum_{i=1}^{N} \left\| \mathbf{p}_i^{(\sigma^{-1}(j))} - \hat{\mathbf{p}}_i^{(j)} \right\|_1. \tag{19}$$

For character classification, we adopt a linear projection to perform character classification. And CTC loss is exploited to address the length inconsistency issue between the ground truth text scripts and predictions:

$$\mathcal{L}_{\text{char}}^{(k)} = \mathbb{1}_{\{k \in \text{Im}(\sigma)\}} \text{CTC}(\mathbf{t}^{\sigma_{-1}(k)}, \hat{\mathbf{t}}_{(k)}). \tag{20}$$

4 Experiments

4.1 Public Datasets

Three public datasets, including Total-Text, ICDAR 2015 (IC15) and CTW1500, are utilized in our experiments. **CTW-1500** is provided by Liu et al [31], focusing on curved text in real-world scenarios. It contains 1,500 images with over 10,000 text annotations. Each text instance is annotated with a 14-point polygon representing curved and irregular shapes.**Total-Text** is an arbitrarily-shaped word-level scene text benchmark released by Ch'ng et al. [32], which consists of 1,255 images for training and 300 for testing with a total of 9,330 annotated text instances. It features a wide range of text orientations, with more than half of the images containing multiple orientations.**ICDAR-2015** benchmark [33] includes 1,000 training images and 500 testing images collected from natural scenes. The dataset features diverse text orientations and complex backgrounds, making it a challenging benchmark for both text localization and recognition tasks.

4.2 Implementation Details

Multi-Query TESTR is implemented using the PyTorch framework. Swin-Transformer is employed as the basic backbone. The shared Transformer encoder applies Deformable DETR. For deformable attention, the number of heads and sampling points is 8 and 4, respectively. The number of proposals N is set to 100. In the shared Transformer decoder, the number of sampled points K is set to 25 by default, unless otherwise specified. Our models predict 37 character-classes on Total-Text and IC15, and 96 character-classes on CTW1500.

For the training phase, data augmentations are utilized, including random rotation with an angle chosen from $[-45°, +45°]$, instance-aware random crop, random resize, and color jitter. Initially, the model is pretrained on SynthText-150k, MLT 2017, and Total-Text for 440k iterations. To mitigate the variance across different datasets, the model is finetuned on the corresponding dataset prior to evaluation.

All experiments are performed on 8 Nvidia GeForce RTX 3090 GPUs and an Ubuntu 20.04 workstation with an Intel(R) Xeon(R) Silver 4210 2.20GHz CPU and 64GB RAM.

4.3 Comparison With State-of-the-Art Methods

In this section, Multi-Query TESTR is compared with multiple state-of-the-art methods on both arbitrarily-shaped and arbitrarily-oriented scene text benchmarks, which is shown in Table 1 and Table 2.

Method	Backbone	CTW-1500			Total-Text			ICDAR15		
		R	P	F1	R	P	F1	R	P	F1
SegLink	VGG-16	48.4	38.3	42.8	30.3	23.8	36.7	76.8	73.1	75.0
TextSnake	FCN+FPN	85.3	67.9	75.6	74.5	82.7	78.4	73.9	83.2	78.3
LSAE	ResNet-50	77.8	82.7	80.1	-	-	-	-	-	-
TextDragon	VGG-16	82.8	84.5	83.6	85.6	75.7	80.3	-	-	-
TextField	VGG-16	79.8	83.0	81.4	79.9	81.2	80.6	80.1	84.3	82.4
SegLink++	VGG-16	79.8	82.8	81.3	80.9	82.1	81.5	80.3	83.7	82.0
CRAFT	ResNet-50	81.1	86.0	83.5	79.9	87.6	83.6	84.3	89.8	86.9
PAN	ResNet-50	81.2	86.4	83.7	81.0	89.3	85.0	81.9	84.0	82.9
DRRG	ResNet-50	83.0	85.9	84.5	84.9	86.5	85.9	84.7	88.5	86.6
CentripetalText	ResNet-50	79.9	88.3	83.9	82.5	90.5	86.3	-	-	-
DMPNet	ResNet-50	61.7	63.9	62.7	-	-	-	-	-	-
CTD+TLOC	ResNet-50	69.8	77.4	73.4	71.0	74.0	73.0	77.1	84.5	80.6
EAST	ResNet-50	49.7	78.7	60.4	50.0	36.2	42.0	78.3	83.3	80.7
Textboxes++	ResNet-50	-	-	-	-	-	-	78.5	87.8	82.9
RRPN	ResNet-50	-	-	-	-	-	-	77.0	84.0	80.0
ATTR	ResNet-50	80.2	80.1	80.1	76.2	80.9	78.5	83.3	90.4	86.8
MSR	ResNet-50	79.0	84.1	81.5	73.0	85.2	78.6	-	-	-
PSENet-1s	ResNet-50	79.7	84.8	82.2	78.0	84.0	80.9	85.5	88.7	87.1
LOMO	ResNet-50	76.5	85.7	80.8	79.3	87.6	83.3	83.5	91.3	87.2
Mask TTD	ResNet-50	79.0	79.7	79.4	74.5	79.1	76.7	87.6	86.6	87.1
Counter-Net	ResNet-50	84.1	83.7	83.9	83.9	86.9	85.4	86.1	87.6	86.9
DB	ResNet-50	80.2	86.9	83.4	82.5	87.1	84.7	82.7	88.2	85.4
Mask TextSpotter	ResNet-50	-	-	-	82.4	88.3	85.2	87.3	86.6	87.0
Boundary	ResNet-50-FPN	-	-	-	88.9	85.0	87.0	-	-	-
PCR	ResNet-50	82.3	87.2	84.7	82.0	88.5	85.2	-	-	-
FCE-Net	ResNet-50	83.4	87.6	85.5	82.5	89.3	85.8	-	-	-
TBTD	ResNet-50	-	-	-	-	-	-	78.3	89.8	83.7
ACE	ResNet-50	-	-	-	-	-	-	82.6	82.1	82.4
TextBPN	ResNet-50	80.6	87.7	84.0	83.3	90.8	86.9	-	-	-
GLASS	ResNet-50	-	-	-	85.5	90.8	88.1	-	-	-
SwinTextSpotter	Swin-T-FPN	-	-	88.0	-	-	88.0	-	-	-
ABCNet v2	ResNet-50-FPN	83.8	85.6	84.7	84.1	90.2	87.0	86.0	90.4	88.1
TESTR	ResNet-50	83.1	89.7	87.1	81.4	93.4	86.9	89.7	90.3	90.0
DeepSolo	ResNet-50	-	-	-	82.1	93.1	87.3	87.4	92.8	90.0
Multi-Query TESTR	ResNet-50	-	-	**88.1**	84.5	92.2	88.2	88.5	92.2	90.2
Multi-Query TESTR	Swin-T	-	-	**88.4**	85.6	92.7	**89.1**	90.5	92.0	91.2

Table 1: Performance Comparison of State-of-the-art Scene Text Detection Methods on CTW-1500, Total-Text, and ICDAR15 Datasets.

Method	Total-Text		CTW-1500		ICDAR 2015		
	None	Full	None	Full	S	W	G
Mask TextSpotter [10]	65.3	77.4	-	-	83.0	77.7	73.5
FOTS [20]	-	-	21.1	39.7	83.6	79.1	65.3
TextDragon [11]	48.8	74.8	39.7	72.4	82.7	78.5	73.0
Text Perceptron [21]	69.7	78.3	57.0	-	80.5	76.6	65.1
ABCNet [12]	64.2	75.7	45.2	74.1	-	-	-
Mask TextSpotter v3 [16]	71.2	78.4	-	-	83.3	78.1	74.2
PGNet [22]	63.1	-	-	-	83.3	78.3	63.5
MANGO [23]	72.9	83.6	58.9	78.7	85.4	80.1	73.9
ABCNet v2 [13]	70.4	78.1	57.5	77.2	82.7	78.5	73.0
PAN++ [24]	68.6	78.6	-	-	82.7	78.2	69.2
Boundary [25]	65.0	76.1	46.1	73.0	82.5	77.4	71.7
SwinTextSpotter [26]	74.3	84.1	51.8	77.0	83.9	77.3	70.5
SRSTS [27]	78.8	86.3	-	-	85.6	81.7	74.5
TPSNet [28]	78.5	84.1	60.5	80.1	-	-	-
GLASS [29]	79.9	86.2	-	-	84.7	80.1	76.3
TESTR [7]	73.3	83.9	56.0	81.5	85.2	79.4	73.6
TTS [6]	78.2	86.3	-	-	85.2	81.7	77.4
ABINet++ [30]	77.6	84.5	60.2	80.3	84.1	80.4	75.4
DeepSolo [8]	79.7	87.0	60.1	78.4	88.0	83.5	79.1
Multi-Query TESTR (ours)	**81.9**	**88.4**	**62.2**	80.5	**89.4**	**85.2**	**80.6**

Table 2: Performance Comparison of State-of-the-art Text Recognition Methods on Total-Text, CTW-1500, and ICDAR 2015 Datasets.

4.3.1 Multi-Query TESTR for Arbitrarily Shaped Scene Text. As mentioned above, the SCUT-CTW1500 and Total-Text datasets focus on arbitrarily shaped text instances. We conducted experiments on the SCUT-CTW1500 and Total-Text datasets. From Table 1 and Table 2, Multi-Query TESTR achieves detection results of 88.4% and 89.1% on the SCUT-CTW1500 and Total-Text datasets, respectively, demonstrating excellent detection performance. Naturally, the outstanding detection results also provide favorable conditions for the recognition task. Similarly, Multi-Query TESTR also achieves impressive recognition results on the Total-Text dataset, with a recognition accuracy of 81.9% without a provided lexicon and 88.4% with a provided lexicon. Compared to methods that train detection and recognition tasks separately (such as TTS[6], TESTR[7],DeepTTS[34]), Multi-Query TESTR employs a shared encoder to output multi-scale query features and the Transformer decoder performs predictions through task-aware reasoning and interaction on shared decoding branches or parallel subtask decomposition branches, thereby enhancing the model's detection and recognition performance.

Compared to TTS[6], Multi-Query TESTR improves the detection results by 1.3%. Compared with TTS, our proposed Multi-Query TESTR is designed to handle multiple tasks, including a shared-parameter encoder and a fused Transformer decoder by replacing the dual decoders in TESTR. As a result, Multi-Query TESTR can process complex text more efficiently. Therefore, from Table 2, on the Total-Text dataset, Multi-Query TESTR achieves superior recognition results compared to TTS[6], with an improvement of 3.7% without a provided lexicon and 2.1% with a provided lexicon.

TESTR[7] achieves detection results of 87.1% and 86.9% on the SCUT-CTW1500 and Total-Text datasets, respectively. Compared to TESTR[7], our method, Multi-Query TESTR, improves the detection results by 1.3% on the SCUT-CTW1500 dataset and by 2.2% on the Total-Text dataset. Compared with TESTR, Multi-Query TESTR employs a shared encoder to output multi-scale query features, and the Transformer decoder performs predictions through task-aware reasoning and interaction on shared decoding branches or parallel subtask decomposition branches. Notably, we are pleasantly surprised to find that, on the Total-Text dataset without a provided lexicon, Multi-Query TESTR improves the recognition results by 8.6% compared to TESTR[7] (81.9% vs. 73.3%).

4.3.2 Multi-Query TESTR for Multi-Oriented Scene Text. For the arbitrarily oriented dataset ICDAR-2015, Multi-Query TESTR also outperforms other methods. During the training phase, we employed data augmentation techniques, including random flipping, instance-aware random

Method	Backbone	Detection			End-to-End		FPS
		P	R	F	None	Full	
FOTS	ResNet-50	52.3	38.0	44.0	32.2	-	-
Textboxes	ResNet-50-FPN	62.1	45.5	52.5	36.3	48.9	1.4
Mask TextSpotter	ResNet-50-FPN	69.0	55.0	61.3	52.9	71.8	4.8
CharNet	ResNet-50-Hourglass57	87.3	85.0	86.1	66.2	-	1.2
Text Dragon	VGG-16	85.6	75.7	80.3	48.8	74.8	-
Boundary TextSpotter	ResNet-50-MSF	88.9	85.0	87.0	65.0	76.1	-
Unconstrained	ResNet-50-FPN	83.3	83.4	83.3	67.8	-	-
Text Perceptron	ResNet-50-FPN	88.8	81.8	85.2	69.7	78.3	-
Mask TextSpotter v3	ResNet-50-FPN	-	-	-	71.2	78.4	-
ABCNet-MS	ResNet-50-FPN	-	-	-	69.5	78.4	6.9
ABCNet v2	ResNet-50-FPN	90.2	84.1	87.0	70.4	78.1	10
MANGO	ResNet-50-FPN	-	-	-	72.9	83.6	4.3
PGNet	ResNet-50-FPN	85.5	86.8	86.1	63.1	-	35.5
TESTR-Bezier	ResNet-50	92.8	83.7	88.0	71.6	83.3	5.5
TESTR-Polygon	ResNet-50	93.4	81.4	86.9	73.3	83.9	5.3
Multi-Query TESTR	ResNet-50	92.2	84.5	**88.2**	**81.9**	**88.4**	12

Table 3: Scene text spotting results on Total-Text.

Dataset	Run 1	Run 2	Run 3	Run 4	Run 5	Mean ± Std.
ICDAR 2015	86.1	86.0	86.2	86.3	85.9	86.10±0.12%
Total-Text	88.4	88.5	88.3	88.6	88.2	88.40±0.15%

Table 4: Stability analysis of Multi-Query TESTR over five independent runs.

cropping, random scaling, and color jittering. As shown in Table 1, Multi-Query TESTR achieves the highest recognition results on the ICDAR-2015 dataset under three different lexicon settings, particularly reaching 89.4% under the Strong (S) setting.

DeepSolo[8] uses a single decoder to simultaneously perform detection and recognition tasks through shared or task-aware queries. It implicitly or explicitly describes the interaction between tasks through shared queries. Considering the synergy and unique characteristics of text detection and recognition, we adopt a multi-query strategy to handle the relationship between the two subtasks. Multi-Query TESTR improves detection results by 1.8% on the Total-Text dataset compared to DeepSolo (89.1% vs. 87.3%). Additionally, on the Total-Text dataset, our method achieves significantly better recognition results than DeepSolo[8]. From Table 2, on the Total-Text dataset without a provided lexicon, our recognition results improve by 2.2%, and with a provided lexicon, they improve by 0.7%.

To evaluate the computational efficiency of the proposed single-decoder architecture, we report the inference speed (FPS) in Table 3. Our Multi-Query TESTR achieves an inference speed of 12.0 FPS, which is a significant improvement over the dual-decoder framework TESTR. This performance gain is attributed to our unified Multi-Query Transformer Decoder, which eliminates the redundant computation associated with parallel dual-branch decoding. Compared to other state-of-the-art methods, our model maintains a superior balance between spotting accuracy and inference throughput.

In addition, we emphasize the connection of different subtask query embeddings to explore the common attributes of diverse element contents. Although existing scene text detection and recognition models have already achieved impressive performance, Multi-Query TESTR demonstrates even more outstanding results on the SCUT-CTW1500 and Total-Text datasets, proving its effectiveness in handling arbitrarily shaped text instances.

To verify the reliability of the performance improvements, we conducted a stability analysis by training the Multi-Query TESTR five times with different initialization seeds. On the ICDAR 2015 dataset, our framework achieved a stable H-mean of 86.1% with a standard deviation of only ±0.12%. A similar trend was observed on the Total-Text dataset, where the variance remained within ±0.15%. The detailed results of these five independent runs are reported in Table 4. These results indicate that our Concate-Separate-Collaborative (CSC) paradigm provides a consistent

Multi-Task	Specific-Task	Swin-T	Detection			Recognition	
			P	R	F1	None	Full
✗	✗	✗	93.4	81.4	86.9	73.3	83.9
✓	✗	✗	92.6	84.2	88.2	78.0	86.5
✓	✓	✗	92.2	84.5	88.2	80.9	87.5
✓	✓	✓	92.7	85.6	**89.1**	**81.9**	**88.4**

Table 5: Ablation experiment results on the Total-Text dataset

optimization landscape, ensuring that the reported gains are statistically significant rather than the result of training variance.

4.4 Ablation Study

To evaluate the effectiveness of each component in Multi-Query TESTR, this section conducts extensive ablation experiments, primarily focusing on the multi-task graph learning layer and task-specific query learning layers. By analyzing the results of these ablation experiments, we validate the rationality and necessity of each component, summarized in Table 5.

4.4.1 Effectiveness of Multi-Task Query Learning: From Table 5, we observe that incorporating the multi-task graph learning layer into the baseline model significantly improves the performance of both detection and recognition results. Interestingly, the detection accuracy alone is enhanced by 1.3% through the sole addition of our proposed MTGL layer. This improvement primarily stems from the exploration and utilization of task-specific attributes: by constructing detection task graphs and recognition task graphs, the model extracts task-related relational information from task-specific query embeddings, explores contextual dependencies between tasks, and generates graph embeddings for multi-task representations. These embeddings enable cross-task reasoning, thereby maximizing mutual influences among tasks. Notably, whether in multi-task, specific-task, or their combined scenarios, it is evident that integrating vocabulary tables yields substantially better results compared to their exclusion. More surprisingly, this component not only enhances the model's expressive power and generalization capabilities but also improves its understanding and processing of tasks by leveraging inter-task relational information. Consequently, the model achieves remarkable performance gains in complex real-world scenarios.

4.4.2 Effectiveness of Specific-Task Query Learning: After mining the correlations between tasks in the multi-task graph learning layer, the next step is to consider the extraction and utilization of task-related information. Specifically, the task-specific query embedding learning layer, which includes the detection task query learning layer and the recognition task query learning layer, can effectively extract task-related features and integrate them into multi-task prediction. As shown in the third row of Table 5, it is surprising that when the baseline model combines the multi-task graph learning layer with the task-specific query learning layer, the detection performance improves from 88.2% (only multi-task) to 88.8%. This indicates that extracting task-related features is of great significance for improving model performance. Moreover, when using all three parts, the detection and recognition performance reaches the highest value of 89.1% and 88.4%.

As can be seen in Fig. 3 and Fig. 4, qualitative detection and recognition results of Multi-Query TESTR are presented. We collect visualization sample results from the TotalText dataset and the CTW1500 dataset, which contain arbitrarily shaped and oriented datasets. For Figure 3(a-1) the characters "CRAFT" and "RESTAURANT" are accurately detected and recognized semantically by our Multi-Query TESTR. However,Figure 3(b-1) only detected the text instances,and the falsely detected Chinese character produces a wrong recognition result with the TESTR method. The English text spotter is not designated to detect and recognize Chinese character sets at present. The redundantly detected polygons should not be expected. With Collaborative Content-to-Query Learning, our Multi-Query TESTR predicts a relative lower score for unexpected Chinese instance classification. Semantic-specific queries and collaborative queries exert interaction impact on the classification head. Using two decoders does not achieve the same effectiveness as a single decoder. This may be due partly to the direct enhancement of network performance through the synergy between detection and recognition, and partly because the queries learned from two decoders may generate unexpected heterogeneity. For Figure 4(a-1) and Figure 4(b-1), the overlapped "ST" are accurately detected by our model, while TESTR models

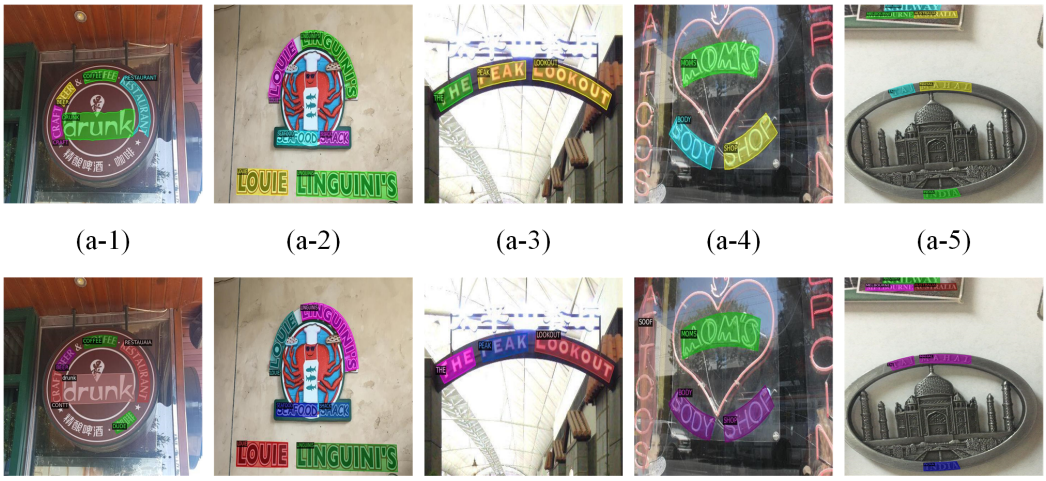

Figure 3: Qualitative detection and recognition results predicted by both our Multi-Query TESTR and TESTR method[7] on multiple selected samples from Total-Text.

Figure 4: Qualitative detection and recognition results predicted by both our Multi-Query TESTR and TESTR method[7] on multiple selected samples from SCUT-CTW1500.

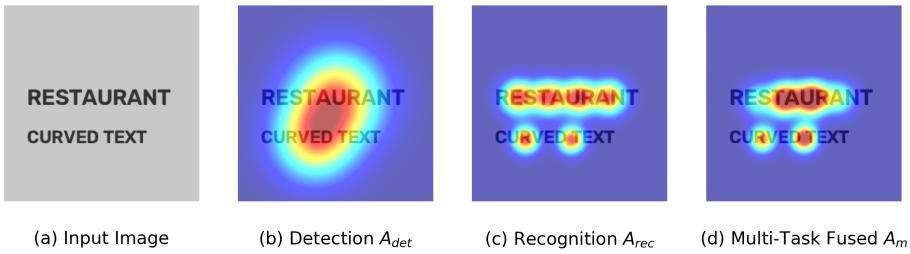

Figure 5: Visualization of task interaction and mutual learning. (a) Input image containing text instances. (b) Detection-specific attention heatmap ($\mathbf{A}_{det}$) before MTGL fusion, showing diffuse spatial activation. (c) Recognition-specific semantic heatmap ($\mathbf{A}_{rec}$), focusing on individual characters. (d) The collaborative multi-task heatmap ($\mathbf{A}_m$) after MTGL, demonstrating how semantic confidence actively refines and sharpens the spatial localization boundaries.

suffer from missing characters. And "ASSOCIATION" is correctly detected and recognized. Curved text instances with various font size and style can be handled properly. The observation in visualization is consistent with the performance comparison shown in the metrics in Table 2. It is evident that using multi-queries is able to achieve excellent detection and recognition performance. This method performs well in spotting texts against various complex backgrounds.

To fully isolate the contribution of each component, as demonstrated in Table 5, we observe that the Specific-Task Query Learning alone struggles to capture the global context (achieving an F1 of 86.9% on the baseline). Conversely, relying solely on Multi-Task Query Learning captures the synergy but dilutes the fine-grained geometric precision needed for detection. It is only through the explicit combination of both within our CSC paradigm that the model reaches its peak performance (89.1% F1). This confirms that the components are highly complementary; the MTGL provides the macro-level dependencies, while the specific queries ensure micro-level accuracy.

4.4.3 Interpretability of Task Interaction. To intuitively demonstrate the effectiveness of the proposed Concate-Separate-Collaborative (CSC) paradigm and the mutual learning mechanism, we provide qualitative visual evidence of the query interactions. As emphasized in our methodology, the Multi-Task Graph Learning (MTGL) module facilitates a Mutual Gradient Gating effect between the detection and recognition branches.

Figure 5 visualizes the cross-attention heatmaps of the task-specific queries before and after the multi-task adjacency matrix fusion ($\mathbf{A}_m$).

As shown in Figure 5(b), relying solely on the detection query ($\mathbf{A}_{det}$) leads to a relatively diffuse and coarse attention map that covers the general text region but lacks precision at the boundaries. Conversely, as illustrated in Figure 5(c), the recognition query ($\mathbf{A}_{rec}$) generates strong, focused semantic activations for specific character sequences.

By fusing these matrices via element-wise multiplication ($\mathbf{A}_m = \mathbf{A}_{det} \odot \mathbf{A}_{rec}$), the high-confidence semantic features actively refine the spatial boundaries (Figure 5(d)). The fused heatmap explicitly shows that the recognition task does not passively wait for detection crops; instead, it proactively guides the detection branch to focus precisely on valid character regions while suppressing surrounding background noise. This synergistic case study firmly supports our claim that the Multi-Query TESTR inherently learns to make the detection and recognition tasks mutually beneficial, rather than processing them as isolated parallel branches.

4.5 Limitations and Failure Cases Analysis

Despite the state-of-the-art performance achieved by Multi-Query TESTR on multiple challenging benchmarks, the model still exhibits certain limitations in extreme scenarios. Through a qualitative error analysis, we have identified two primary types of failure cases. First, in instances of severe occlusion or extreme illumination variance, the detection branch occasionally fails to construct a complete spatial graph, leading to fragmented text proposals or truncated recognition results. Second, the model can sometimes be misled by ambiguous background textures that share high structural similarities with character strokes (e.g., intricate architectural lines, grid-like fences, or specific clothing patterns). In such cases, the recognition branch may generate false-positive queries. While our Collaborative Content-to-Query Learning mitigates this issue by fusing multi-scale feature contents, completely resolving these edge cases remains a challenge. Future

work will explore integrating more explicit background-suppression attention mechanisms and leveraging extensive pre-training on synthetically occluded datasets to further push the boundaries of the model's robustness.

5 Conclusion

In this paper, we have presented Multi-Query TESTR, a novel Multi-Query TExt Spotter TRansformer. By applying a Shared Transformer Encoder and a Multi-Query Transformer Decoder, our approach effectively integrates control points regression for text detection and semantic understanding for text recognition within a unified network. The spatial-aware query learning for control points regression and semantic-aware query learning for recognition are jointly optimized through cross task attention, enabling the model to reason dependencies and learn mutual effects between detection and recognition tasks. The introduction of Collaborative Multi-Scale Deformable Cross Attention further enhances the interaction between queries and feature contents, producing outstanding detection and recognition performance.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62471272, 61806107 and 62201314, the Opening Project of State Key Laboratory of Digital Publishing Technology.

References

[1] S. Long, X. He, C. Yao, Scene text detection and recognition: The deep learning era, International Journal of Computer Vision 129 (01 2021).

[2] X.-F. Wang, Z.-H. He, K. Wang, Y.-F. Wang, L. Zou, Z.-Z. Wu, A survey of text detection and recognition algorithms based on deep learning technology, Neurocomputing 556 (2023) 126702.

[3] Q. Ye, D. S. Doermann, Text detection and recognition in imagery: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (2015) 1480–1500.

[4] D. Peng, X. Wang, Y. Liu, J. Zhang, M. Huang, S. Lai, J. Li, S. Zhu, D. Lin, C. Shen, X. Bai, L. Jin, Spts: Single-point text spotting, in: Proceedings of the 30th ACM International Conference on Multimedia, MM '22, Association for Computing Machinery, New York, NY, USA, 2022, pp. 4272–4281.

[5] Y. Liu, J. Zhang, D. Peng, M. Huang, X. Wang, J. Tang, C. Huang, D. Lin, C. Shen, X. Bai, L. Jin, Spts v2: Single-point scene text spotting, arXiv preprint arXiv:2301.01635 (2024).

[6] Y. Kittenplon, I. Lavi, S. Fogel, Y. Bar, R. Manmatha, P. Perona, Towards weakly-supervised text spotting using a multi-task transformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 4604–4613.

[7] X. Zhang, Y. Su, S. Tripathi, Z. Tu, Text spotting transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 9519–9528.

[8] M. Ye, J. Zhang, S. Zhao, J. Liu, T. Liu, B. Du, D. Tao, Deepsolo: Let transformer decoder with explicit points solo for text spotting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 19348–19357.

[9] R. Hu, A. Singh, Unit: Multimodal multitask learning with a unified transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 1439–1449.

[10] P. Lyu, M. Liao, C. Yao, W. Wu, X. Bai, Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 67–83.

[11] W. Feng, W. He, F. Yin, X.-Y. Zhang, C.-L. Liu, Textdragon: An end-to-end framework for arbitrary shaped text spotting, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9076–9085.

[12] Y. Liu, H. Chen, C. Shen, T. He, L. Jin, L. Wang, Abcnet: Real-time scene text spotting with adaptive bezier-curve network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9809–9818.

[13] Y. Liu, C. Shen, L. Jin, T. He, P. Chen, C. Liu, H. Chen, Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (11) (2021) 8048–8064.

[14] M. Huang, J. Zhang, D. Peng, H. Lu, C. Huang, Y. Liu, X. Bai, L. Jin, Estextspotter: Towards better scene text spotting with explicit synergy in transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 19495–19505.

[15] Y. Xu, X. Li, H. Yuan, Y. Yang, L. Zhang, Multi-task learning with multi-query transformer for dense prediction, IEEE Transactions on Circuits and Systems for Video Technology 34 (2) (2023) 1228–1240.

[16] M. Liao, G. Pang, J. Huang, T. Hassner, X. Bai, Mask textspotter v3: Segmentation proposal network for robust scene text spotting, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16, Springer, 2020, pp. 706–722.

[17] H. Bi, C. Xu, C. Shi, G. Liu, Y. Li, H. Zhang, J. Qu, Srrv: A novel document object detector based on spatial-related relation and vision, IEEE Transactions on Multimedia 25 (2023) 3788–3798.

[18] H. Bi, C. Xu, C. Shi, G. Liu, H. Zhang, Y. Li, J. Dong, Hgr-net: Hierarchical graph reasoning network for arbitrary shape scene text detection, IEEE Transactions on Image Processing (2023).

[19] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: European Conference on Computer Vision, Springer, 2020, pp. 213–229.

[20] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, J. Yan, Fots: Fast oriented text spotting with a unified network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5676–5685.

[21] L. Qiao, S. Tang, Z. Cheng, Y. Xu, Y. Niu, S. Pu, F. Wu, Text perceptron: Towards end-to-end arbitrary-shaped text spotting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 11899–11907.

[22] P. Wang, C. Zhang, F. Qi, S. Liu, X. Zhang, P. Lyu, J. Han, J. Liu, E. Ding, G. Shi, Pgnet: Real-time arbitrarily-shaped text spotting with point gathering network, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 2782–2790.

[23] L. Qiao, Y. Chen, Z. Cheng, Y. Xu, Y. Niu, S. Pu, F. Wu, Mango: A mask attention guided one-stage scene text spotter, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 2467–2476.

[24] W. Wang, E. Xie, X. Li, X. Liu, D. Liang, Z. Yang, T. Lu, C. Shen, Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (9) (2021) 5349–5367.

[25] H. Wang, P. Lu, H. Zhang, M. Yang, X. Bai, Y. Xu, M. He, Y. Wang, W. Liu, All you need is boundary: Toward arbitrary-shaped text spotting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 12160–12167.

[26] M. Huang, Y. Liu, Z. Peng, C. Liu, D. Lin, S. Zhu, N. Yuan, K. Ding, L. Jin, 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, 2022, pp. 4593–4603.

[27] J. Wu, P. Lyu, G. Lu, C. Zhang, W. Pei, Single shot self-reliant scene text spotter by decoupled yet collaborative detection and recognition, arXiv preprint arXiv:2207.07253 (2022).

[28] W. Wang, Y. Zhou, J. Lv, D. Wu, G. Zhao, N. Jiang, W. Wang, Tpsnet: Reverse thinking of thin plate splines for arbitrary shape scene text representation, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 5014–5025.

[29] R. Ronen, S. Tsiper, O. Anschel, I. Lavi, A. Markovitz, R. Manmatha, Glass: Global to local attention for scene-text spotting, in: European Conference on Computer Vision, Springer, 2022, pp. 249–266.

[30] S. Fang, Z. Mao, H. Xie, Y. Wang, C. Yan, Y. Zhang, Abinet++: Autonomous, bidirectional and iterative language modeling for scene text spotting, IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (6) (2022) 7123–7141.

[31] L. Yuliang, J. Lianwen, Z. Shuaitao, Z. Sheng, Detecting curve text in the wild: New dataset and new solution, arXiv preprint arXiv:1712.02170 (2017).

[32] C. K. Ch'ng, C. S. Chan, Total-text: A comprehensive dataset for scene text detection and recognition, in: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Vol. 1, IEEE, 2017, pp. 935–942.

[33] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. L. et al., Icdar 2015 competition on robust reading, in: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), IEEE, 2015, pp. 1156–1160.

[34] Y. Xie, C. Xu, C. Shi, J. Li, Z. Yuan, Q. Qiao, Deeptts: Enhanced transformer-based text spotter via deep interaction between detection and recognition tasks, in: Pacific Rim International Conference on Artificial Intelligence, Springer, 2024, pp. 450–462.