MSTAR: Box-Free Multi-Query Scene Text Retrieval with Attention Recycling

Liang Yin **Xudong Xie** **Zhang Li** **Xiang Bai** **Yuliang Liu***

Huazhong University of Science and Technology

{liangyin, xdxie, zhangli, xbai, ylliu}@hust.edu.cn

Abstract

Scene text retrieval has made significant progress with the assistance of accurate text localization. However, existing approaches typically require costly bounding box annotations for training. Besides, they mostly adopt a customized retrieval strategy but struggle to unify various types of queries to meet diverse retrieval needs. To address these issues, we introduce Multi-query Scene Text retrieval with Attention Recycling (MSTAR), a box-free approach for scene text retrieval. It incorporates progressive vision embedding to dynamically capture the multi-grained representation of texts and harmonizes free-style text queries with style-aware instructions. Additionally, a multi-instance matching module is integrated to enhance vision-language alignment. Furthermore, we build the Multi-Query Text Retrieval (MQTR) dataset, the first benchmark designed to evaluate the multi-query scene text retrieval capability of models, comprising four query types and $16k$ images. Extensive experiments demonstrate the superiority of our method across seven public datasets and the MQTR dataset. Notably, MSTAR marginally surpasses the previous state-of-the-art model by 6.4% in MAP on Total-Text while eliminating box annotation costs. Moreover, on the MQTR benchmark, MSTAR significantly outperforms the previous models by an average of 8.5%. The code and datasets are available at https://github.com/yingift/MSTAR.

1 Introduction

Scene text appears almost everywhere in daily life and is an essential ingredient for image-text searching [6]. Traditional scene text retrieval [9] typically aims to search images based on word or phrase queries and could benefit applications such as handwritten signature retrieval [54, 53] and key frame extraction [35]. However, real-life retrieval needs could be diverse. For instance, people often search for an article with several key words, which cannot be fulfilled with a single word. Additionally, non-ocr visual semantics is also vital for scene text searching. In this work, we study the multi-query scene text retrieval that aims to handle queries of various types within a unified model. This could facilitate applications such as searching disambiguation [31, 16] and visual document indexing [48, 4].

Off-the-shelf scene text retrieval methods [9, 38, 39] achieve text retrieval by explicitly localizing the text and matching the query with scene text instances, as shown in Fig. 1 (a). A straightforward solution is the text spotting methods [22, 20, 49]. They first spot the text in images and match it using edit distance. However, the two divided stages could cause error accumulation between text spotting and query matching. To mitigate this, detection-based methods [38, 41, 39] represent detected ROIs in dense embedding, integrating text detection and similarity learning into an end-to-end framework.

*Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

 https://doi.org/10.52202/085713-1939

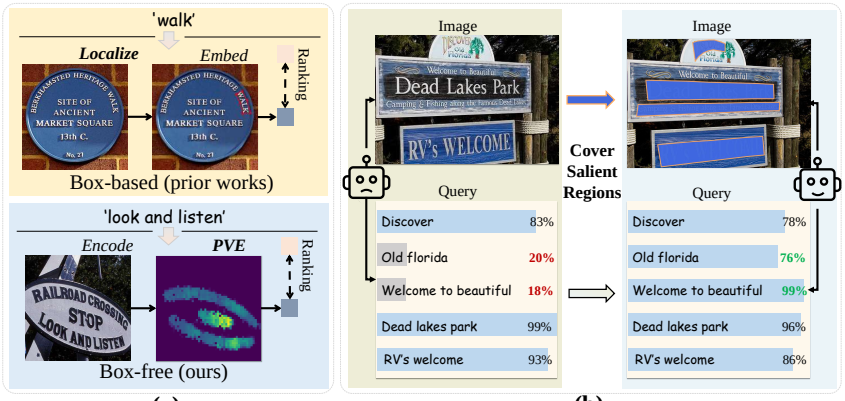


Figure 1: (a) MSTAR achieves scene text retrieval without the aid of box annotations. (b) Image-text matching experiments with VLM [19]. Detailed text instances like "welcome to beautiful" and "old florida" in the image receive lower matching scores. While manually covering salient text regions which receive the higher scores, the model can adaptively recognize the detailed text.

Recently, FDP [51] leverages bounding boxes to guide CLIP [30] in focusing on text regions to achieve accurate retrieval. Despite these advancements, they require expensive box annotations for training. As different retrieval tasks require varying labels, it is very costly to obtain multiple types of bounding boxes, i.e., word-level, text-line, and common object bounding boxes.

Recently, the large-scale box-free pre-training of Vision-Language Models (VLMs) [18, 52] has shown impressive capability on various tasks [1, 55, 50]. To apply box-free methods for scene text retrieval, we conduct image-text matching experiments, as shown in Fig. 1 (b). The observations reveal that box-free models tend to overlook detailed text instances. Moreover, manually masking salient text regions mitigates this issue by enabling the model to adaptively capture image details. More observations are supplemented in the appendix. Inspired by these, we introduce a box-free model termed Multi-query Scene Text retrieval with Attention Recycling (MSTAR). It starts with leveraging pre-trained VLMs for scene text retrieval. To better capture detailed scene text features, the progressive vision embedding is designed to shift attention from salient regions to insalient areas with less attention iteratively. To support diverse retrieval queries, text queries are encoded by the multi-modal encoder with style-aware instructions. A multi-instance matching module is then designed to establish the cross-modal alignment. In this way, MSTAR seamlessly unifies diverse text queries and aligns image-query embeddings without the need of any positional supervision.

To evaluate the performance of multi-query retrieval, we have carefully built a Multi-Query Text Retrieval (MQTR) benchmark. Beyond traditional word and phrase queries, this dataset introduces two additional, valuable retrieval settings: combined query and semantic query. The combined query comprises several discontinuous key texts (words or phrases) to enable more precise retrieval. Semantic queries, on the other hand, are image descriptions that require understanding both scene text and its non-ocr visual context [45]. In total, the MQTR dataset includes four styles of queries (word, phrase, combined, and semantic) and 16000 images.

Experiments reveal that existing models struggle to simultaneously handle four types of query on the challenging MQTR dataset. In contrast, our proposed MSTAR performs well in multi-query retrieval. Notably, MSTAR impressively surpasses previous work by an average of 8.5% in MAP. Additionally, on seven public retrieval datasets, MSTAR demonstrates competitive performance compared to state-of-the-art box-based methods while significantly reducing annotation costs. Specifically, MSTAR outperforms FDP [51] by 6.4% in MAP on the widely used Total-Text dataset.

We summarize the main contributions as follows: (1) We propose MSTAR, the first box-free method designed for scene text retrieval, which eliminates box annotations and achieves multi-query retrieval. (2) We collect MQTR, a comprehensive benchmark with four types of queries and 16,000 images for evaluating multi-query scene text retrieval. (3) Experiments on seven public datasets and the MQTR benchmark demonstrate the advantages of MSTAR.

2 Related Work

Scene Text Retrieval Datasets. Existing scene text retrieval datasets primarily focus on single-type retrieval such as word and phrase. The IIIT Scene Text Retrieval dataset [27] is a large-scale dataset dedicated to word retrieval comprising 10k images. The COCOText Retrieval dataset is derived from 7k natural images in the COCOText dataset [37]. In addition, several smaller but well-annotated datasets [40, 7] are commonly utilized for evaluating word retrieval performance. The Chinese Street View Text Retrieval dataset [38] contains 23 Chinese queries and 1,667 scene text images from Chinese street views. Besides these word retrieval datasets, the Phrase-level Scene Text Retrieval [51] dataset includes 36 frequently used phrase queries and 1,080 images. The CSVTRv2 [39] dataset supports partial and text-line queries. However, these datasets do not comprehensively support the evaluation of tasks such as combined retrieval and semantic retrieval. In contrast, our MQTR dataset can support four types of query to satisfy diverse needs in real-world applications.

Scene Text Retrieval Methods. Existing methods generally adopt a paradigm that first localize text regions and then match with the query. In the early attempts, Mishra *et al.*[27] proposed identifying approximate character locations and indexing words using spatial constraints. More recent approaches try to integrate the two process into an end-to-end trainable framework. Gomez *et al.*[9] proposed to combine the detected proposals with the Pyramidal Histogram of Characters [2]. Wang *et al.*[38] unified the text detector with a cross-modal similarity model into an end-to-end framework. Its updated version [39] proposed the RankMIL and DPMA algorithms to address the partial scene text retrieval problem. Wen *et al.*[41] transformed cross-modal similarity into uni-modal similarity using image templates. Zeng *et al.*[51] utilized CLIP [30] for scene text retrieval and incorporated box supervision to localize text regions. However, these methods require expensive bounding box annotations for training.

In addition to the specifically designed retrieval methods, text spotters can also be applied to retrieval tasks. They first spot text instances from images and then rank with edit distance. Traditional text spotters with boundary supervision [22, 20, 44] can achieve accurate results. There are also point-supervised [29, 23] and transcription-only supervised methods [36, 42, 43], which yield limited performance. Above all, the separation of text spotting and query matching often leads to error accumulation, and text spotters struggle to handle multi-query retrieval.

Unlike above methods, our MSTAR achieves text retrieval without the bounding-box supervision and harmonize various data labels for multi-query scene text retrieval.

3 Method

3.1 Overview

The overall architecture of MSTAR is depicted in Fig. 2. Given a scene text image, the vision encoder extracts image features f_0, and the Progressive Vision Embedding module progressively captures the scene text features as vision embeddings E_{V}. Simultaneously, text queries are encoded as text embeddings E_{T} by the multi-modal encoder from BLIP-2 [19] with style-aware instructions. The E_{V} and E_{T} are then fed into the Multi-Instance Matching module to align the cross-modal embeddings optimized with contrastive loss. Additionally, a re-ranking process is incorporated by inputting both image features and text queries into the multi-modal encoder to compute one-to-one matching scores. During training, MSTAR is optimized using both contrastive loss and image-text matching loss. For inference, image-text pairs are initially ranked by the cosine similarity, and the top K images are further matched with re-ranking process.

3.2 Progressive Vision Embedding

Vision-Language Models (VLMs) pre-trained on image-caption pairs often focus more on salient visual concepts such as a red circle [33, 47]. However, the detailed and subtle scene text instances typically appear in various insalient regions of natural scenes. As depicted in Fig. 1 (b), the "old florida" and "welcome to beautiful" are overlooked as the model focus on more salient regions. This problem leads to a high miss rate for small text scenarios in scene text retrieval tasks. To mitigate this, we propose a Progressive Vision Embedding (PVE) approach to extract visual embeddings.

 https://doi.org/10.52202/085713-1939

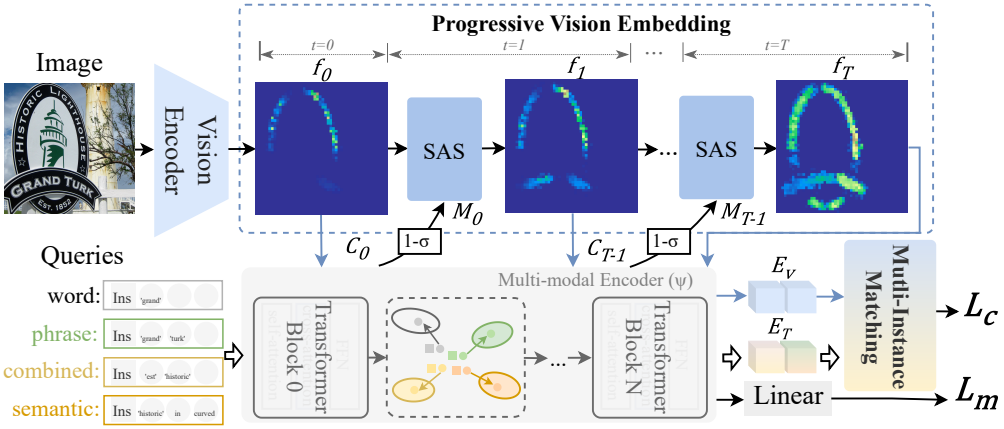


Figure 2: Overview of MSTAR. MSTAR is built upon four key components: a vision encoder ϕ, the Progressive Vision Embedding (PVE), the multi-modal encoder ψ, and the multi-instance matching module (MIM). PVE incorporates image features f_{t} and the mask M_{t} derived from cross-attention map C_t, progressively shifting attention from salient features to fine-grained regions.

Given a scene text image $I \in \mathbb{R}^{\mathrm{H}\times\mathrm{W}\times 3}$, the vision encoder ϕ encodes I into initial image features, denoted as $f_0 \in \mathbb{R}^{\mathrm{L}\times\mathrm{D}}$. Then a two-layer MLP is used to align the hidden dimensions of ϕ and the multi-modal encoder ψ stacked with transformer blocks. Subsequently, ψ is leveraged to capture the scene text vision embeddings from f_0 with learnable queries $Q_{\mathrm{I}} \in \mathbb{R}^{\mathrm{Q}\times\mathrm{d}}$. The initial vision embeddings are denoted as $E^0_{\mathrm{V}} \in \mathbb{R}^{\mathrm{Q}\times\mathrm{d}}$.

Since the model tends to focus on salient image features (visualized in Fig. 2), E^0_{V} struggles to fully capture the detailed text instance representation. To force the model to focus on less salient features, we propose the Salient Attention Shift (SAS) module. Motivated by observations in Sec. 1, the SAS uses a mask-attention layer to automatically shift image attention. Unlike previous methods [5] that use ground truth as supervision, the mask in our approach is derived from the cross-attention layers of the multi-modal encoder ψ. Concretely, we first calculate the mean of the cross-attention weights of ψ as the cross-attention map $C_{\mathrm{t\text{-}1}}$. Then $C_{\mathrm{t\text{-}1}}$ is binarized with a binarization algorithm σ and inverted to obtain a binary mask $M_{\mathrm{t\text{-}1}}$. This is formulated in Eq. 1.

$$M_{\mathrm{t\text{-}1}} = 1 - \sigma(C_{\mathrm{t\text{-}1}}), \tag{1}$$

where the σ consists of a thresholding with low threshold for coarse filtering, a marker-based watershed algorithm for precise binarization, and a connected components algorithm to avoid over-segmentation. In the t-th step, the SAS refines image features as follows:

$$f_{\mathrm{t}} = \mathcal{S}(f_{\mathrm{t\text{-}1}}, \mathrm{mask} = M_{\mathrm{t\text{-}1}})), \tag{2}$$

where $\mathcal{S}$ denotes multi-head self-attention, $f_{\mathrm{t\text{-}1}}$ is image features and $M_{\mathrm{t\text{-}1}}$ is the binary mask. For each pixel $M^{\mathrm{i,j}}_{\mathrm{t\text{-}1}}$, a value of 0 indicates that the corresponding image features of the previous iteration received high attention, while a value of 1 indicates lower attention. With $M_{\mathrm{t\text{-}1}}$, the SAS learns to adaptively reduce the weight of salient features and focus more on the neglected features.

In each iteration, the SAS dynamically renews the image features, as illustrated in Fig. 2. Then the multi-modal encoder ψ embeds the f_{t} to $E^{\mathrm{t}}_{\mathrm{V}}$. Then the vision embeddings $\{E^0_{\mathrm{V}}, E^1_{\mathrm{V}}, \ldots E^{\mathrm{T}}_{\mathrm{V}}\}$ are concatenated to $E_{\mathrm{V}} \in \mathbb{R}^{(\mathrm{T}+1)\mathrm{Q}\times\mathrm{d}}$, where T is the maximum recurrent steps.

3.3 Instruction Aware Text Representation

In unified training of multi-query scene text retrieval, the difference of query styles (e.g., format and characteristics) could cause semantic discrepancy. For example, the phrase query contains several continuous words with linguistic semantics, but the combined query contains several discrete key words. To harmonize the representation of these text queries, a short style-aware text instruction is introduced to guide the embedding for each type of query. The process is formulated as follows:

$$E_{\mathrm{T}} = \psi(\mathrm{Concate}[T_{\mathrm{i}}, T_{\mathrm{Q}}]), \tag{3}$$

where ψ denotes the multi-modal encoder, T_{i} represents instructions and T_{Q} denotes text queries. The instructions prompt the ψ to distinguish query types. As shown in Fig. 2, queries of each style are encoded into different representation space during training. To speed up training, all of the text queries (words, phrases, combined, and semantic queries) paired to the image are encoded altogether. The ψ encodes the queries into text embeddings $E_{\mathrm{T}} \in \mathbb{R}^{\mathrm{N}\times \mathrm{d}}$ which consist of single-word embeddings $E_{\mathrm{w}} \in \mathbb{R}^{N_{\mathrm{w}}\times \mathrm{d}}$ and multi-word embeddings $E_{\mathrm{m}} \in \mathbb{R}^{N_{\mathrm{m}}\times \mathrm{d}}$. The N is the total number of text queries paired to the image. The N_{w} and N_{m} denote the number of single-word queries and multi-word queries, respectively.

3.4 Multi-Instance Matching

After obtaining the vision embeddings $E_{\mathrm{V}} \in \mathbb{R}^{(\mathrm{T+1})\mathrm{Q}\times \mathrm{d}}$ and text embeddings $E_{\mathrm{T}} \in \mathbb{R}^{\mathrm{N}\times \mathrm{d}}$, the problem is to build the one-to-one alignment for the multi-type and multi-instance embeddings. The previous study either aggregate the multiple vision embeddings into one embedding [19] or adopts the late interaction [15, 8]. However, due to the implicit matching mechanism, these strategies need massive training for vision-language alignment.

To mitigate these, we propose the Multi-Instance Matching (MIM) module to explicitly assign the one-to-one matching relations for vision-language embeddings. MIM comprises two parallel branches for processing single-word and multi-word queries, respectively. In the single-word branch, the Hungarian matching algorithm [17] is exploited to explicitly assign the one-to-one matching relation between $E_{\mathrm{w}} \in \mathbb{R}^{N_{\mathrm{w}}\times \mathrm{d}}$ and $E_{\mathrm{V}} \in \mathbb{R}^{(\mathrm{T+1})\mathrm{Q}\times \mathrm{d}}$. Specifically, we first construct a cosine similarity matrix of size $N_{\mathrm{w}} \times$ (T+1)Q. Since N_{w} is typically unequal to (T+1)Q, we pad the matrix with zeros to create a square matrix. Finally, the first N_{w} rows of the results are used for one-to-one correspondences.

In the multi-word branch, since the multi-word queries contain abundant semantic information, a light-weight cross-attention layer is used to aggregate the vision features under text constraint in the second branch. This process is formulated as Eq. 4.

$$E_{\mathrm{vt}} = \mathcal{F}((\mathcal{C}(\mathcal{Q} = E_{\mathrm{w}}, \mathcal{K}, \mathcal{V} = E_{\mathrm{V}}))), \tag{4}$$

where $\mathcal{F}$ denotes a feed-forward network and $\mathcal{C}$ denotes multi-head cross-attention. The two branches adaptively shift to cope with different types of queries, i.e., word retrieval relies solely on the first branch, while multi-word queries with the second. Thanks to this flexible alignment approach, mixed training data labels can be effectively leveraged for training multi-query retrieval models.

3.5 Optimization

MSTAR is optimized with both contrastive learning and image-text matching task. Contrastive learning enables the model to separately encode vision embeddings and text embeddings. A dual contrastive loss $\mathcal{L}_{\mathrm{c}}$ aligns the vision and text embeddings.

$$\mathcal{L}_{\mathrm{c}} = \alpha\mathcal{L}_{\mathrm{t2v}} + \mathcal{L}_{\mathrm{v2t}}, \tag{5}$$

where α is a hyperparameter to maintain the numerical approximation equivalence of the two losses. Since the number of queries is usually greater than the number of images, α is set to 1.5 in our implementation. The image-text matching process simultaneously feeds image features and text queries into the multi-modal encoder ψ. An image-text matching score is then computed using a linear layer with a two-cell output. This score is optimized using a cross-entropy matching loss $\mathcal{L}_{\mathrm{m}}$. The overall loss is the sum of $\mathcal{L}_{\mathrm{c}}$ and $\mathcal{L}_{\mathrm{m}}$.

4 Experiments

4.1 Multi-Query Text Retrieval Dataset

To comprehensively evaluate the performance of models on multi-query scene text retrieval, we carefully build the Multi-Query Text Retrieval (MQTR) dataset. The MQTR dataset includes four sub-tasks: word, phrase, combined, and semantic retrieval. The construction of this dataset leverages well-annotated public datasets [37, 21, 13, 14, 7, 34, 24, 51], along with images obtained from Google Image Search. The word, phrase, and combined subsets each contain 5,000 images and the 200 most frequently occurring queries. The semantic subset consists of 1,000 images and 25 queries collected from the web. The semantic subset was manually collected with 10-15 positive images and an equal

 https://doi.org/10.52202/085713-1939

Dataset	Venue	Word	Phrase	Combined	Semantic	Q. Num	Images
Total-Text [7]	IJDAR'20	✓	✗	✗	✗	60	300
CTW [21]	PR'19	✓	✗	✗	✗	100	500
IC15 [14]	ICDAR'15	✓	✗	✗	✗	100	500
CTR [37]	Arxiv'16	✓	✗	✗	✗	500	7196
STR [9]	ECCV'18	✓	✗	✗	✗	50	10k
CSVTR [38]	CVPR'21	✗	✓	✗	✗	23	1667
PSTR [51]	ACM MM'24	✗	✓	✗	✗	36	1080
MQTR	-	✓	✓	✓	✓	625	16k

Table 1: Statistics of public scene text retrieval datasets and our MQTR dataset in terms of query types, number of queries, and number of images.

Method	Venue	AVG.	Word	Phrase	Combined	Semantic
Box Based						
ABCNet [22]	TPAMI'21	24.13	26.14	15.15	36.47	18.74
MaskTextSpotter [20]	ECCV'20	32.43	46.72	27.53	29.08	26.37
TDSL [38]	CVPR'21	58.25	69.11	40.83	72.71	50.36
Deepsolo [49]	CVPR'23	52.04	67.54	25.68	72.14	42.79
TG-Bridge [11]	CVPR'24	54.09	69.89	30.21	**75.53**	40.73
Box Free						
SPTSv2 [23]	TPAMI'23	35.18	33.56	21.24	50.76	35.16
BLIP-2 [19]	PMLR'23	36.13	17.31	32.76	25.80	68.63
SigLIP [52]	CVPR'23	36.06	17.81	32.88	21.81	72.23
BLIP-2 (FT) [19]	PMLR'23	58.11	58.09	42.23	60.84	71.24
MSTAR	-	**66.78**	**73.27**	**44.22**	74.48	**75.14**

Table 2: Evaluations of MAP% on MQTR. FT denotes fine-tune. The best results are shown in **bold**.

number of hard negative samples for each query. The hard negatives samples refer to images that contain three types of objects: (1) visual elements with semantics similar to the query (e.g., an apple and the word "apple"), (2) text instances with similar shapes, and (3) text instances with similar meanings. The inclusion of hard negatives poses additional challenges by introducing visually and textually confounding samples, thus assessing the capacity of retrieval models to distinguish visual semantics from OCR-based semantics. As demonstrated in Tab. 1, our MQTR dataset is the first comprehensive benchmark to support four types of query in scene text retrieval. You can refer to the appendix for more construction details and images samples from the datasets.

4.2 Implementation details

The visual encoder ϕ is initialized from ViT-Base-512 of SigLIP [52]. The multi-modal encoder ψ is initialized from BLIP-2 [19]. The number of query tokens Q_1 is set to 64 with interpolation, which is consistent with the setting of the vanilla BLIP-2 in our comparison experiments. The weights of the MLP, SAS, and MIM modules are randomly initialized. The MSTAR model was trained on four NVIDIA A800 GPUs and evaluated on a single GPU, using the AdamW optimizer [25]. A multi-stage training is adopted with progressive resolution increasing from 512×512, 640×640, to 800×800. For re-ranking, the top 2% of images are selected from the initial retrieval results. For instance, in a dataset containing $10k$ images, only the top 200 images are used for re-ranking.

4.3 Multi-Query Scene Text Retrieval

To enable multi-query retrieval, we collect a training dataset consisting of $95k$ images. First, $50k$ synthetic images with word transcriptions are leveraged from SynthText-900k [9]. Then $20k$ real images containing captions are collected from TextCap [34]. We use labels with both image captions and text transcriptions acquired with Rosetta [3]. In addition, we have

BLIP-2 [19]	TDSL [38]	SigLIP [52]	FDP [51]	MSTAR
85.49	89.40	89.56	92.28	**95.71**

Table 3: Comparisons on Phrase-level Scene Text Retrieval dataset [51].

Method	Venue	SVT	STR	CTR	Total-Text	CTW	IC15	Avg.	FPS
Box Based									
Mishra *et al.* [27]	ICCV'13	42.70	56.24	-	-	-	-	-	0.1
Jaderberg *et al.* [12]	IJCV'16	86.30	66.50	-	-	-	-	-	0.3
Gomez *et al.* [9]	ECCV'18	83.74	69.83	41.05	-	-	-	-	43.5
Mafla *et al.* [26]	PR'21	85.74	71.67	-	-	-	-	-	42.2
TDSL [38]	CVPR'21	89.38	77.09	66.45	74.75	59.34	77.67	74.16	12.0
Wang *et al.* [39]	TPAMI'24	-	81.02	**72.95**	-	-	-	-	9.3
Wen *et al.* [41]	WSDM'23	90.95	77.40	-	80.09	-	-	-	11.0
FDP-RN50×16 [51]	ACM MM'24	89.63	**89.46**	-	79.18	-	-	-	11.8
Box Free									
BLIP-2 (FT) [19]	PMLR'23	88.73	85.40	45.75	77.20	82.33	55.13	72.42	37.2
MSTAR	-	**91.31**	86.25	60.13	85.55	90.87	81.21	82.56	14.2
MSTAR (+re-rank)	-	91.11	86.14	65.25	**86.96**	**92.95**	**82.69**	**84.18**	6.9

Table 4: Comparisons with scene text retrieval methods of MAP% on 6 public word retrieval datasets. The best results are shown in **bold**, and the second results are underlined.

Method	Venue	SVT	STR	CTR	Total-Text	CTW	IC15	Avg.	FPS
Box Based									
ABCNet [22]	TPAMI'21	82.43	67.25	41.25	73.23	74.82	69.28	68.04	17.5
MaskTextspotterV3 [20]	ECCV'20	83.14	74.48	55.54	83.29	80.03	77.00	75.58	2.4
Deepsolo [49]	CVPR'23	87.15	76.58	67.22	83.19*	87.67*	82.80*	80.77	10.0
TG-Bridge [11]	CVPR'24	87.23	81.30	**70.08**	**87.11***	88.39*	**83.55***	82.94	6.7
Box Free									
SPTSv2 [23]	TPAMI'23	78.08	62.11	48.39	73.61*	83.30*	66.27*	68.63	7.6
MSTAR	-	**91.31**	**86.25**	60.13	85.55	90.87	81.21	82.56	14.2
MSTAR (+re-rank)	-	91.11	86.14	65.25	86.96	**92.95**	82.69	**84.18**	6.9

Table 5: Comparisons with mainstream scene text spotting methods. * indicates that the model has been fine-tuned on the corresponding training set. The best results are highlighted in **bold**, and the second results are underlined.

synthesized $25k$ images with phrase transcriptions using the synthesis engine [10]. Additionally, word or phrase annotations are utilized to form nonrepeated combined queries for images containing over one text instance. More training details can be found in the appendix.

On the MQTR dataset, we perform evaluation with both the representative box-based models and box-free models for multi-query scene text retrieval. For text spotting methods, we use the normalized edit distance to measure query-image matching scores following [38]. To evaluate the models that can only handle word queries, we calculate the mean similarity of each word as the image-text scores for multi-word queries. The codes and weights are acquired from their official repositories.

Evaluation on multi-query scene text retrieval. As the results reported in Tab. 2, box-based methods [38, 49, 11] typically perform better on word queries and combined queries, which demand fine-grained scene text perception. However, these box-based models cannot leverage the rich linguistic semantics for phrase and semantic retrieval. Compared to box-based methods, MSTAR outperforms TG-Bridge[11] by 3.38% and Deepsolo[49] by 5.73% in word retrieval. On the other hand, VLMs such as BLIP-2 (ViT-L) [19] and SigLIP (ViT-B-512) [52] perform better on phrase and semantic queries but struggle in word and combined retrieval. Compared to them, MSTAR outperforms SigLIP by 11.34% on phrase retrieval and 2.91% on semantic retrieval. In terms ofms of overall results, our MSTAR obtains an improvement of 8.53% over previous works on average. These comparison results show the great advantages of our MSTAR on multi-query retrieval.

Additionally, MSTAR is also evaluated on the phrase-level scene text retrieval dataset [51]. As shown in the Tab. 3, our MSTAR achieves 95.71% of MAP on the benchmark.

https://doi.org/10.52202/085713-1939

Ins	MIM	PVE	CTR	SVT	STR	Total-Text	CTW	IC15	MQTR
✗	✗	✗	52.87	90.07	81.57	82.32	87.28	76.71	65.79
✓	✗	✗	54.65	90.70	82.81	83.19	88.96	77.15	66.15
✓	✓	✗	55.77	91.02	85.00	84.01	90.31	79.23	65.69
✓	✓	✓	60.13	91.31	86.25	85.55	90.87	81.21	66.78

Table 6: Ablation studies on instruction, multi-instance matching, progressive vision embedding, denoted as Ins, MIM, PVE, respectively.

4.4 Word-level Scene Text Retrieval

In this part, we present comparison experiments on word-level retrieval. The model is trained on $100k$ images randomly sampled from SynthText-900k [9] and 5k images from MLT-5K [28] dataset. The evaluation setting keeps the same as the previous method [38]. Note that Deepsolo [49], TG-Bridge [11], and SPTSv2 [23] are well fine-tuned on the training set of Total-Text, CTW and IC15 dataset.

Comparisons with text retrieval methods. We conduct comprehensive evaluations on the test sets of six public datasets, the results are presented in Tab. 4. Compared to FDP-RN50$\times$16 [51], our MSTAR achieves an improvement of 1.68% on SVT and 6.37% on Total-Text. While MSTAR demonstrates a slightly lower performance on the STR dataset, it eliminates the cost of expensive bounding-box for training. Compared to TDSL [38], our method significantly outperforms the method across five datasets. Notably, our MSTAR surpasses TDSL by 9.16% on STR, 10.80% on TotalText, and 31.53% on CTW. These results verify the robust capabilities of our model. However, on the CTR dataset that contains extreme small text, our method underperforms TDSL. This is due to the absence of precise box supervision of our method, which is a common problem for box-free methods. Overall, MSTAR outperforms TDSL by an average 8.40% in MAP across six datasets. To further enhance performance, we re-rank the top 2% of retrieved images by jointly feeding the text queries and images into the model. This re-ranking strategy improves performance by an additional 1.56% in MAP.

Comparisons with text spotting methods. Tab. 5 shows the comparison results with scene text spotting methods. Compared to box-free method, our model significantly outperforms SPTSv2 with an average improvement of 13.93% in MAP. To further verify the advantages of MSTAR, we compare it with state-of-the-art box-based text spotting methods. The results in Tab. 5 indicate that MSTAR achieves competitive performance with the advanced TG-Bridge on average. Moreover, MSTAR offers over twice the inference speed compared to TG-Bridge (14.2 FPS vs. 6.7 FPS) due to the absence of the text detection module. These results show that our model achieves competitive performance with leading fully supervised models while eliminating the bounding box for training.

4.5 Ablation Study

To validate the effectiveness of each component, comprehensive ablation studies were conducted.

Ablation study on three core components. The overall results are reported in Tab. 6. We begin by directly training the vision encoder, MLP and multi-modal encoder with standard contrastive learning. The results suggest that this baseline performs poorly on the fine-grained recognition capability, especially on the CTR and IC15 datasets which features small text. Then instructions are adopted to prompt the model to encode queries which leads to improvement. Subsequently, the MIM is added, which leads to a significant improvement of 2.19% on STR and 1.35% on CTW. However, a slight performance drop of 0.46% is observed on the MQTR dataset. This may be because of the use of Hungarian matching, which is effective for word-level queries and instance-level alignment, introduces confusion when handling more complex queries.

Lastly, we incorporate PVE which is designed to improve the retrieval performance of the insalient objects such as small and detailed text instances. The results show a substantial improvement on CTR (4.36%) and IC15 (1.98%), validating the effectiveness of PVE in small text scenarios.

Ablation study on the binary σ algorithm. We investigate three variants of σ, as the results reported in Tab. 7. 1) Zero Padding: A binary mask with all zero values is used, which means the mask does not constrain the image attention. The results show only a slight improvement on CTR (0.91%), which is probably due to the effectiveness of the progressive representation strategy. 2) The second

σ	CTR	Total-Text	IC15
No PVE	55.76	84.01	79.23
Zero Pad	56.67	83.79	79.27
TH+CC	59.66	85.17	80.17
TH+WS+CC	60.13	85.55	81.21

Table 7: Ablation studies on σ algorithm. TH denotes ThresHolding, CC denotes Connected Components, and WS denotes WaterShed algorithm.

choice of σ is composed of thresholding and connected components. This variant can produce a mask to guide the SAS module to refine attention focus. However, it requires repeated tuning of the thresholds. 3) The third choice is to first apply coarse filtering to the background with a low threshold and then obtain precise binarization results using the watershed algorithm. This approach eliminates the need for complex hyperparameter tuning and achieves substantial improvements, including 4.37% on CTR, 1.54% on Total-Text dataset, and 1.98% on the IC15 dataset. We adopt the third variant for our model.

Ablation study on the number of iteration steps T in PVE. We investigate the impact of the number of recurrent steps T in PVE. As the results presented in Tab. 8, performance improves significantly as T increases from 0 to 1. As T increases from 1 to 3, we can also observe noticeable improvement on the three datasets. Since the inference speed decreases with the recurrent steps increases, we adopt T=1 for the final model to balance efficiency and effectiveness.

T	CTR	Total-Text	CTW	FPS
0	55.76	84.01	90.31	16.5
1	60.13	85.55	90.87	14.2
2	60.47	86.68	90.95	12.9
3	60.87	87.66	91.24	11.2

Table 8: The impact of the recurrent steps T.

5 Discussion

(a) 'dream big' tshirt **(b) 'lighthouse'** **(c) '1888', 'celtic'** **(d) 'restaurant'** **(e) 'copyright'**

Figure 3: Visualization of the text localization of our MSTAR. The image shows the localization of (a) semantic, (b) phrase, and (c) combined query, as well as (d) curved and (e) dense word instances.

Application of MSTAR for text localization. To further validate the effectiveness of MSTAR, we use it for text localization using Grad-CAM [32]. As illustrated in Fig. 3, MSTAR can localize different types of queries. For example, MSTAR accurately identifies the target text instance "dream big" on a t-shirt, distinguishing it from the "dream big" of the shorts in Fig. 3 (a). In addition, MSTAR can also accurately localize curved and dense text instances. As Fig. 3 (e) shows, MSTAR successfully identifies the word "copyright" within a document page image. These results demonstrate that MSTAR can accurately localize text instances without box supervision.

Discussion about the pre-trained model. We discuss how the pre-trained model affects performance on the scale of parameters and data. We tested different variants of CLIP [46], BLIP [18], BLIP-2 [19], and SigLIP [52] on the CTR dataset. The results show that despite the massive pre-training, the models struggle to achieve retrieval on small text instances. Details are provided in the appendix.

Limitations. Although our method shows evident improvements in fine-grained scene text retrieval, it still lags behind box-based methods when handling extremely small and dense text instances. This

https://doi.org/10.52202/085713-1939

limitation stems from insufficient positional supervision, which is a common limitation to box-free approaches.

6 Conclusion

This work investigates the fundamental OCR task of scene text retrieval via a novel box-free paradigm. It incorporates PVE to shift visual attention to fine-grained scene text. Our model demonstrates competitive retrieval performance with state-of-the-art box-based methods while significantly reducing annotation cost. Moreover, for the first time, we study the multi-query scene text retrieval enabling broader real-world applications. Experiments show that neither box-based methods nor general cross-modal retrievers can handle such a challenging task effectively, while our method can serve as a strong baseline for this task. This study is expected to inspire future research on weakly supervised pre-training for visual document foundation models.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (Grant 2022YFC3301703) and the NSFC (Grants 62206104, 62225603).

References

[1] Aviad Aberdam, David Bensaïd, Alona Golts, Roy Ganz, Oren Nuriel, Royee Tichauer, Shai Mazor, and Ron Litman. Clipter: Looking at the bigger picture in scene text recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21706–21717, 2023.

[2] Jon Almazán, Albert Gordo, Alicia Fornés, and Ernest Valveny. Word spotting and recognition with embedded attributes. *IEEE transactions on pattern analysis and machine intelligence*, 36(12):2552–2566, 2014.

[3] Fedor Borisyuk, Albert Gordo, and Viswanath Sivakumar. Rosetta: Large scale system for text detection and recognition in images. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 71–79, 2018.

[4] Davide Caffagni, Federico Cocchi, Nicholas Moratelli, Sara Sarto, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Wiki-llava: Hierarchical retrieval-augmented generation for multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1818–1826, 2024.

[5] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022.

[6] Mengjun Cheng, Yipeng Sun, Longchao Wang, Xiongwei Zhu, Kun Yao, Jie Chen, Guoli Song, Junyu Han, Jingtuo Liu, Errui Ding, et al. Vista: Vision and scene text aggregation for cross-modal retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5184–5193, 2022.

[7] Chee-Kheng Ch'ng, Chee Seng Chan, and Cheng-Lin Liu. Total-text: toward orientation robustness in scene text detection. *International Journal on Document Analysis and Recognition (IJDAR)*, 23(1):31–52, 2020.

[8] Manuel Faysse, Hugues Sibille, Tony Wu, Gautier Viaud, Céline Hudelot, and Pierre Colombo. Colpali: Efficient document retrieval with vision language models. *arXiv preprint arXiv:2407.01449*, 2024.

[9] Lluís Gómez, Andrés Mafla, Marçal Rusinol, and Dimosthenis Karatzas. Single shot scene text retrieval. In *Proceedings of the European conference on computer vision (ECCV)*, pages 700–715, 2018.

[10] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2315–2324, 2016.

[11] Mingxin Huang, Hongliang Li, Yuliang Liu, Xiang Bai, and Lianwen Jin. Bridging the gap between end-to-end and two-step text spotting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15608–15618, 2024.

[12] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading text in the wild with convolutional neural networks. *International journal of computer vision*, 116:1–20, 2016.

[13] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. In *2013 12th international conference on document analysis and recognition*, pages 1484–1493. IEEE, 2013.

[14] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th international conference on document analysis and recognition (ICDAR)*, pages 1156–1160. IEEE, 2015.

[15] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48, 2020.

[16] Anastasia Kritharoula, Maria Lymperaiou, and Giorgos Stamou. Large language models and multimodal retrieval for visual word sense disambiguation. *arXiv preprint arXiv:2310.14025*, 2023.

[17] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2 (1-2):83–97, 1955.

[18] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[19] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[20] Minghui Liao, Guan Pang, Jing Huang, Tal Hassner, and Xiang 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*, pages 706–722. Springer, 2020.

[21] Yuliang Liu, Lianwen Jin, Shuaitao Zhang, Canjie Luo, and Sheng Zhang. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90:337–345, 2019.

[22] Yuliang Liu, Chunhua Shen, Lianwen Jin, Tong He, Peng Chen, Chongyu Liu, and Hao 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):8048–8064, 2021.

[23] Yuliang Liu, Jiaxin Zhang, Dezhi Peng, Mingxin Huang, Xinyu Wang, Jingqun Tang, Can Huang, Dahua Lin, Chunhua Shen, Xiang Bai, et al. Spts v2: single-point scene text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[24] Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Icdar 2023 competition on hierarchical text detection and recognition. In *International Conference on Document Analysis and Recognition*, pages 483–497. Springer, 2023.

[25] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[26] Andrés Mafla, Ruben Tito, Sounak Dey, Lluís Gómez, Marçal Rusinol, Ernest Valveny, and Dimosthenis Karatzas. Real-time lexicon-free scene text retrieval. *Pattern Recognition*, 110:107656, 2021.

[27] Anand Mishra, Karteek Alahari, and CV Jawahar. Image retrieval using textual cues. In *Proceedings of the IEEE international conference on computer vision*, pages 3040–3047, 2013.

[28] Nibal Nayef, Yash Patel, Michal Busta, Pinaki Nath Chowdhury, Dimosthenis Karatzas, Wafa Khlif, Jiri Matas, Umapada Pal, Jean-Christophe Burie, Cheng-lin Liu, et al. Icdar2019 robust reading challenge on multi-lingual scene text detection and recognition—rrc-mlt-2019. In *2019 International conference on document analysis and recognition (ICDAR)*, pages 1582–1587. IEEE, 2019.

[29] Dezhi Peng, Xinyu Wang, Yuliang Liu, Jiaxin Zhang, Mingxin Huang, Songxuan Lai, Jing Li, Shenggao Zhu, Dahua Lin, Chunhua Shen, et al. Spts: single-point text spotting. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4272–4281, 2022.

 https://doi.org/10.52202/085713-1939

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[31] Brigit Schroeder and Subarna Tripathi. Structured query-based image retrieval using scene graphs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 178–179, 2020.

[32] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[33] Aleksandar Shtedritski, Christian Rupprecht, and Andrea Vedaldi. What does clip know about a red circle? visual prompt engineering for vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11987–11997, 2023.

[34] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: a dataset for image captioning with reading comprehension. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 742–758. Springer, 2020.

[35] Hao Song, Hongzhen Wang, Shan Huang, Pei Xu, Shen Huang, and Qi Ju. Text siamese network for video textual keyframe detection. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 442–447. IEEE, 2019.

[36] Jingqun Tang, Su Qiao, Benlei Cui, Yuhang Ma, Sheng Zhang, and Dimitrios Kanoulas. You can even annotate text with voice: Transcription-only-supervised text spotting. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4154–4163, 2022.

[37] Andreas Veit, Tomas Matera, Lukas Neumann, Jiri Matas, and Serge Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. *arXiv preprint arXiv:1601.07140*, 2016.

[38] Hao Wang, Xiang Bai, Mingkun Yang, Shenggao Zhu, Jing Wang, and Wenyu Liu. Scene text retrieval via joint text detection and similarity learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4558–4567, 2021.

[39] Hao Wang, Minghui Liao, Zhouyi Xie, Wenyu Liu, and Xiang Bai. Partial scene text retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[40] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *2011 International conference on computer vision*, pages 1457–1464. IEEE, 2011.

[41] Lilong Wen, Yingrong Wang, Dongxiang Zhang, and Gang Chen. Visual matching is enough for scene text retrieval. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 447–455, 2023.

[42] Lilong Wen, Xiu Tang, and Dongxiang Zhang. Twist: Text-only weakly supervised scene text spotting using pseudo labels. In *Proceedings of the 2024 International Conference on Multimedia Retrieval*, pages 275–284, 2024.

[43] Jingjing Wu, Zhengyao Fang, Pengyuan Lyu, Chengquan Zhang, Fanglin Chen, Guangming Lu, and Wenjie Pei. Wecromcl: Weakly supervised cross-modality contrastive learning for transcription-only supervised text spotting. In *European Conference on Computer Vision*, pages 289–306. Springer, 2025.

[44] Xudong Xie, Ling Fu, Zhifei Zhang, Zhaowen Wang, and Xiang Bai. Toward understanding wordart: Corner-guided transformer for scene text recognition. In *European conference on computer vision*, pages 303–321. Springer, 2022.

[45] Xudong Xie, Liang Yin, Hao Yan, Yang Liu, Jing Ding, Minghui Liao, Yuliang Liu, Wei Chen, and Xiang Bai. Pdf-wukong: A large multimodal model for efficient long pdf reading with end-to-end sparse sampling. *arXiv preprint arXiv:2410.05970*, 2024.

[46] Chuhui Xue, Wenqing Zhang, Yu Hao, Shijian Lu, Philip HS Torr, and Song Bai. Language matters: A weakly supervised vision-language pre-training approach for scene text detection and spotting. In *European Conference on Computer Vision*, pages 284–302. Springer, 2022.

[47] Lingfeng Yang, Yueze Wang, Xiang Li, Xinlong Wang, and Jian Yang. Fine-grained visual prompting. *Advances in Neural Information Processing Systems*, 36, 2024.

[48] Xiao Yang, Dafang He, Wenyi Huang, Alexander Ororbia, Zihan Zhou, Daniel Kifer, and C Lee Giles. Smart library: Identifying books on library shelves using supervised deep learning for scene text reading. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 1–4. IEEE, 2017.

[49] Maoyuan Ye, Jing Zhang, Shanshan Zhao, Juhua Liu, Tongliang Liu, Bo Du, and Dacheng 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*, pages 19348–19357, 2023.

[50] Wenwen Yu, Yuliang Liu, Wei Hua, Deqiang Jiang, Bo Ren, and Xiang Bai. Turning a clip model into a scene text detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6978–6988, 2023.

[51] Gangyan Zeng, Yuan Zhang, Jin Wei, Dongbao Yang, Peng Zhang, Yiwen Gao, Xugong Qin, and Yu Zhou. Focus, distinguish, and prompt: Unleashing clip for efficient and flexible scene text retrieval. *arXiv preprint arXiv:2408.00441*, 2024.

[52] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.

[53] Peirong Zhang, Kai Ding, and Lianwen Jin. Capturing more: Learning multi-domain representations for robust online handwriting verification. *arXiv preprint arXiv:2508.01427*, 2025.

[54] Peirong Zhang, Yuliang Liu, Songxuan Lai, Hongliang Li, and Lianwen Jin. Privacy-Preserving Biometric Verification With Handwritten Random Digit String. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 47(4):3049–3066, 2025.

[55] Shuai Zhao, Ruijie Quan, Linchao Zhu, and Yi Yang. Clip4str: A simple baseline for scene text recognition with pre-trained vision-language model. *arXiv preprint arXiv:2305.14014*, 2023.

 https://doi.org/10.52202/085713-1939

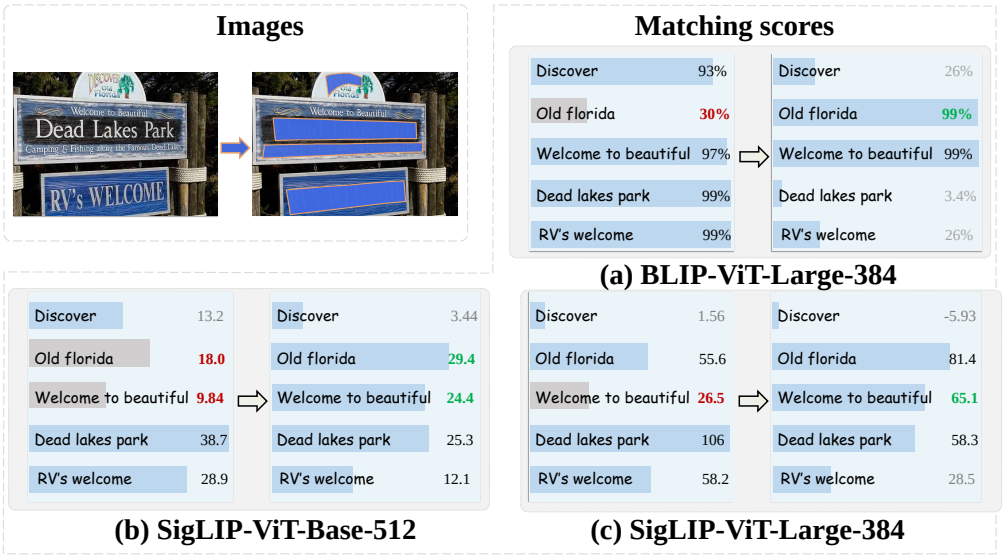


Figure 4: Qualitive analysis of VLMs to process in-salient text instances , which is introduced in Sec. 1, (a) BLIP-ViT-Large-384, (b) SigLIP-ViT-Base-512, and (c) SigLIP-ViT-Large-384.

A Further Analysis of Vision Language Models

In this section, we first provide further analysis of Vision Language Models (VLM) in processing in-salient text instances introduced in Sec. A.1. Second, we present a quantitative evaluation of VLMs on the small-text dataset in Sec. A.2.

A.1 Further Observations on VLMs

To validate the observations that VLMs tend to overlook detailed and insalient text instances, we conducted additional experiments with more models. For the BLIP, we calculate the ITM score which calculates the one-to-one matching probability between the image and the query. For the SigLIP models, we calculate the dot products between the vision and text embeddings without normalization following the official code. As illustrated in Fig. 4 (a), the BLIP can easily capture the salient text like “dead lakes park” but cannot find the text “old florida”. However, it can capture the “old florida” when the salient regions are covered. Similar observations are presented in Fig. 4 (b) (c).

A.2 Performance of VLMs on Small Text Dataset

To test the pre-trained ability of VLMs on fine-grained perception, we evaluate representative VLMs (CLIP [46], BLIP [18], BLIP-2 [19], SigLIP [52]) on the CTR [37] dataset. The CTR dataset is collected from the COCO dataset and features small scene text instances. As the results in Tab. 9, all of the tested VLMs struggle to retrieve images accurately. For instance, the CLIP variants obtain MAP of 8.1% while the BLIP gets 6.9% of MAP. Moreover, increasing model parameters and seen data does not lead to significant performance gains. Among these models, BLIP-2 has the highest MAP of 13.3% of the tested models, indicating limited capacity of VLMs for small text retrieval.

B MQTR dataset

In this section, we first supplement more construction details and annotation procedures for each subset of MQTR dataset in Sec. B.1. We then present representative samples from the MQTR dataset in Sec. B.2.

https://doi.org/10.52202/085713-1939

Model	Parameters	Pre-training Data	MAP%
CLIP-RN50	97M	400M images	6.6
CLIP-ViT-Base	143M	400M Images	6.8
CLIP-ViT-Large	408M	400M Images	8.1
BLIP-ViT-Large	426M	129M Images	6.9
BLIP-2-ViT-Large	452M	129M images	13.3
SigLIP-ViT-Base-512	194M	9B Samples	12.8
SigLIP-ViT-Large-384	622M	9B Samples	11.7
MSTAR-ViT-Base	270M	-	60.13

Table 9: Evaluation of the CLIP-style models on the CoCoText dataset. Parameters denotes the parameters of the model.

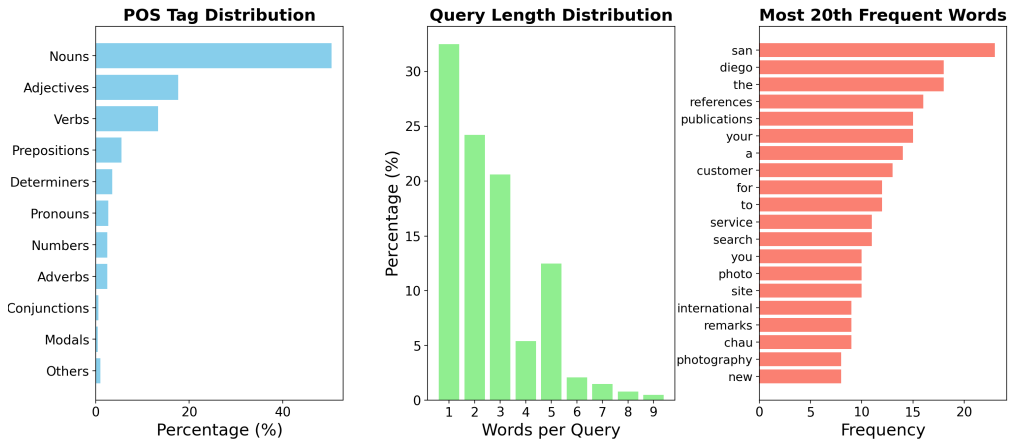


Figure 5: Statistical analysis of the MQTR benchmark.

B.1 Construction details of the MQTR dataset

Word Subset. The word subset includes 5000 images and 200 word queries. The images are sourced from the test set of SVT, CTW, IC15, Total-Text and the CTR. We extract the word-level annotations from the datasets and filter out the words with less than 3 characters (e.g. "st"). After that, 200 word queries are selected according to the word frequency.

Phrase Subset. The images of phrase subset include 1k images from PSTR [51], 1k images manually collected from the Web. Then we use images from HierText [24]. The line-level annotations are used and the lines with only one word are filtered out. Lastly, 200 phrase queries are selected according to frequency.

Combined Subset We first use all images collected from the CTR and HierText dataset. Given the word and line annotations of an image, we then filter out the queries less than 3 characters. Then we implement an DFS algorithm to compute all the combinations that contain 2-4 words. Top 200 text combinations on the dataset are selected according to frequency and repeat ratios. Then images paired to these queries are first selected as positive samples. Then images containing words similar to the 200 combined queries are also selected as negative samples according to the edit distance. Lastly, there are 5000 images for the positive samples and negative samples in total.

Semantic Subset The semantic subset is manually collected from the web. we first brainstorm common-used scene text queries and then search for candidate images from the web. In total, the subset consists of 25 queries and 1000 images.

During the selection of multi-word queries (i.e., phrase and combined types), we manually filter out redundant entries. Specifically, queries that are overly repeated or semantically similar are removed to enhance diversity in the final set.

https://doi.org/10.52202/085713-1939

B.2 Visualization of the MQTR dataset

To clarify the characteristics of the MQTR dataset, we show some examples collected from the google image search engine, as presented in Tab. 10. Since scene text typically appears as fine-grained information but the search engine often recommends the most salient images, we collect more images containing queries in fine-grained concepts. For example, "global weekly" is shown as a section name of "china daily" newspaper, adding more challenges for models. For example, for the query "dream big", the model may have difficulty distinguishing between "bream big" written on the t-shirt and "dream big" written on the shorts.

B.3 Statistics of the MQTR dataset

In total, our dataset contains **625 unique queries** comprising **1,326 words**. As shown in Fig. 5, we report statistics including part-of-speech (POS) tag distribution, query length distribution, and the most frequent words, which collectively demonstrate the diversity of our query set across linguistic and structural dimensions.

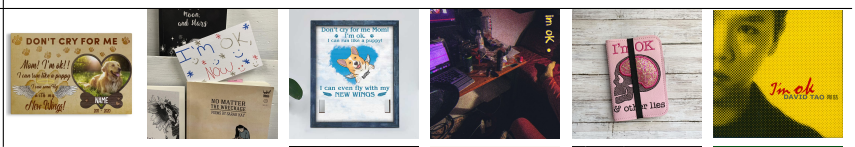
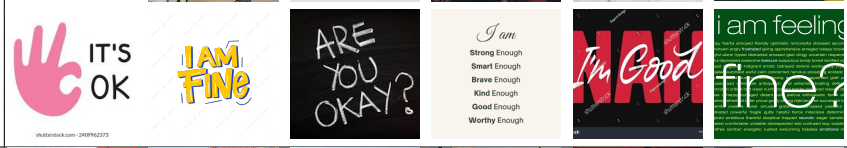
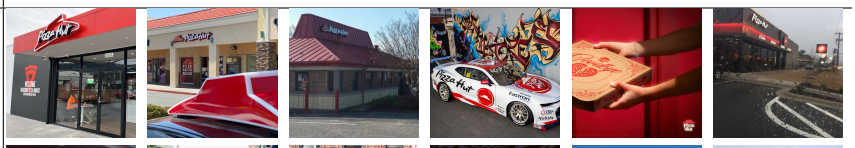
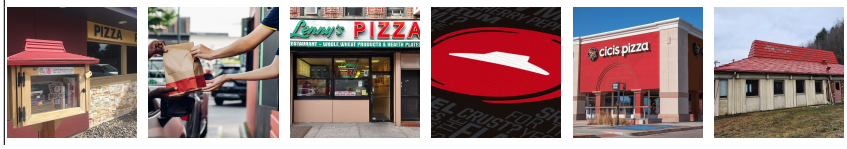
Query		Examples
newspaper of "global weekly"	Positive	
	Negative	
"dream big" tshirt	Positive	
	Negative	
"i'm ok"	Positive	
	Negative	
"pizza hut"	Positive	
	Negative	

Table 10: Examples from the MQTR dataset. The Positive represents the GT images. The Negative denotes hard negative sample described in Sec. 4.

C Experiment Details and Visualization Analysis

C.1 Computational Efficiency Analysis.

Thanks to the design of the PVE module, image features are fed into the ViT only once. Subsequent iterations involve merely the SAS module and the Multi-modal Encoder, both of which are lightweight and thus enable a good trade-off between accuracy and efficiency. We analyze the computational cost and latency on a single A800 GPU, where the *Baseline* removes the PVE module from MSTAR.

Method	GFLOPs	Latency (s)	FPS	CTR
Baseline	248	0.0606	16.5	55.76
MSTAR	310	0.0704	14.2	**60.13**

Table 11: Computation and efficiency comparison on a single A800 GPU.

As shown in Table 11, MSTAR improves accuracy by **4.37%** on the challenging CTR dataset, while achieving an inference speed of **14.2 FPS**, over twice as fast as TG-Bridge (6.7 FPS), and maintaining comparable performance across all six benchmarks.

C.2 Text Region Localization Analysis.

We further evaluate the localization accuracy of the predicted text regions. Binary ground-truth (GT) masks are constructed by extracting the polygon coordinates of all text regions from the CTW dataset. For comparison, we also derive binary masks from the cross-attention maps of BLIP-2 using the same processing pipeline as ours. As reported in Tab. 12, we compute the Intersection over Union (IoU) between each predicted mask and its corresponding GT mask, and report both the mean IoU and the number of high-quality masks with IoU ≥ 0.5 across 500 images.

Method	Mean IoU	High-Quality Masks (IoU $\geq$ 0.5)
BLIP-2	21.78	129 (25.8%)
MSTAR	**50.82**	**304 (60.8%)**

Table 12: Quantitative comparison of text region localization on the CTW dataset. MSTAR produces substantially more accurate and higher-quality text masks than BLIP-2.

C.3 Scale-wise Evaluation on ICDAR2015.

We further evaluate MSTAR on the ICDAR2015 dataset by analyzing performance across different text scales. For each text instance, we compute the ratio between its bounding-box area and the image area using annotations from the original datasets, and group instances into scale intervals as shown in Tab 6.

For each interval, we calculate the AP score for queries whose corresponding text instances fall within the specified scale range. For instance, given the query "apple" and a target scale range $((a, b]]$, we exclude all images where "apple" appears but its area ratio is not within that range. The AP score is then computed using the remaining valid images.

C.4 Training Details

To ensure the reproducibility of our model, we provide the training process and the training hyper parameters, which are reported in Tab. 13. We adopt a progressive training recipe. In the first state, the model is trained on both synthetic data D_{syn} and real data D_{real} at an image resolution of 512. We use a learning rate of 1e-5 and linear cosine schedule with 100 warm-up steps. In this stage, we use a two-layer MLP to align the visual encoder and multi-modal encoder. In the second stage, our model is fine-tuned at the resolution of 640 only on the real data D_{real}. For simplicity, the hyper parameters keep the same with the first stage. In the third stage, we fine-tune the model at a higher resolution of 800 on the real data D_{real}. In the forth stage, the visual encoder is freezed and only the MLP, multi-modal encoder, MIM, and SAS module are optimized at a resolution of 800. For

https://doi.org/10.52202/085713-1939

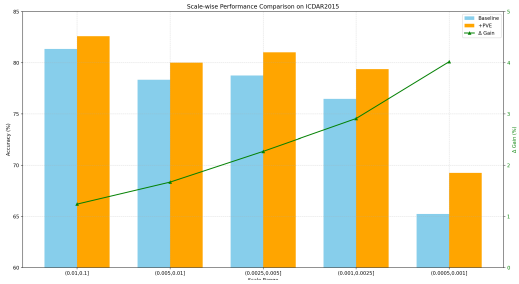


Figure 6: Scale-wise performance comparison on ICDAR2015.

	Phase 1	Phase 2	Phase 3	Phrase 4
Image Resolution	512	640	800	800
Learning Rate	1e-5	1e-5	5e-6	5e-6
Warm-Up steps	100	100	0	0
Freeze ViT	False	False	False	True
Precision of ViT		Float		
Query of ψ		64		
RandomCrop		True		
Dataset	$\{D_{\text{syn}}, D_{\text{real}}\}$	D_{real}	D_{real}	D_{real}

Table 13: Training details of our model.

the word retrieval experiment, the synthetic data D_{syn} refers to 100K images randomly sampled from SynthText-900k, and the real data D_{real} comes from MLT-5K. For the multi-query experiment, D_{syn} includes 50K images randomly sampled from SynthText-900k and 25k images with phrase transcriptions with the synthesis engine. D_{real} is the training set from the TextCap dataset. The labels are the image captions and text transcriptions acquired with Rosetta.

C.5 Experimental Visualization Analysis

We present a qualitative analysis of the retrieval results. As illustrated in Tab. 14, our method can not only effectively leverage linguistic priors for phrase and semantic queries, but also perceive fine-grained scene text instances to achieve word and combined retrieval. For example, MSTAR successfully retrieves the combined query like “reserved” and “some” even when they appear as subtle watermarks in pictures, as depicted in the last row.

Query		Retrieval results		
"speed limit 25"	Bridge [11]			
	MSTAR			
"hydrate or diedrate" is written on a water bottle	Bridge [11]			
	MSTAR			
"may"	BLIP-2 (FT) [19]			
	MSTAR			
"reserved" , "some"	BLIP-2 (FT) [19]			
	MSTAR			

Table 14: Qualitive analysis of the retrieval results. The green boxes mean correct samples and the red boxes denote false images. The queries and images are sampled from the MQTR dataset and CTW [21] dataset.

https://doi.org/10.52202/085713-1939

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction are presented based on the core contribution and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are discussed in Sec. 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The theoretical result are clearly clarified.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We clearly describe the method in Sec. 3, and include the implementation details in Sec. 4.2, evaluation details in Sec. 4.3 and Sec. 4.4. Moreover, we supplement more training details in the appendix. Lastly, we also offer our code and data to ensure reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

https://doi.org/10.52202/085713-1939

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We will provide the code, weights and data in the supplementary files.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We include the implementation details in Sec. 4.2, evaluation details in Sec. 4.3 and Sec. 4.4. Moreover, we supplement more training details in the appendix. C.4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Sure. The evaluation criteria keeps the same with previous studies, which is mentioned in Sec. 4. We conducted extensive experiments on **seven** widely-used public datasets. These experimental results are of statistical significance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We include this in Sec. 4.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research is fully conformed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have claimed the impact of scene text retrieval in Sec. 1, while this research is a step forward to bring a new box-free paradigm to address the task.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

https://doi.org/10.52202/085713-1939

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The model and dataset is commonly used for OCR tasks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited the papers and will respect the licenses.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Do not involve.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

 https://doi.org/10.52202/085713-1939

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: This research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.