Improving Scene Text Retrieval via Stylized Middle Modality

SHIPENG ZHU, JUN FANG, PENGFEI FANG, and HUI XUE, School of Computer Science and Engineering, Southeast University, Nanjing, China and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, Nanjing, China

Scene text retrieval addresses the challenge of localizing and searching for all text instances within scene images based on a query text. This cross-modal task has significant applications in various domains, such as intelligent transportation systems and social media analysis. In practice, ensuring consistency of the same content between two modalities is crucial in improving retrieval accuracy. This article addresses the issue by introducing a stylized middle modality, which fuses the graphical query text with the style of the extracted text proposal. To this end, we propose a stylized middle modality learning (SM^{2}L) framework. The proposed stylized middle modality enables the network to jointly enforce constraints on visual feature coherence and text semantic feature consistency in the optimization phase, thereby minimizing the modality gap in the retrieval space. This brings in two major advantages: (1) SM^{2}L will pave the way to seamlessly benefit the scene text retrieval and (2) the proposed learning paradigm enables the machine to avoid adding redundant computing resources in the inference phase. Substantial experiments demonstrate that the proposed method outperforms the state-of-the-art retrieval performance considerably.

CCS Concepts: • **Computing methodologies → Visual content-based indexing and retrieval**;

Additional Key Words and Phrases: Scene text retrieval, Stylized middle modality, Multi-task learning

ACM Reference format:
Shipeng Zhu, Jun Fang, Pengfei Fang, and Hui Xue. 2024. Improving Scene Text Retrieval via Stylized Middle Modality. *ACM Trans. Multimedia Comput. Commun. Appl.* 20, 12, Article 379 (November 2024), 18 pages.
https://doi.org/10.1145/3696209

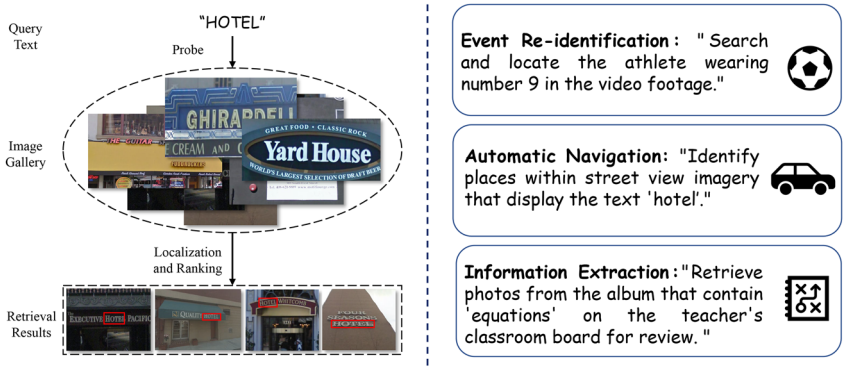
This work was supported by the National Natural Science Foundation of China (Nos. 62476056, 62076062, and 62306070) and the Social Development Science and Technology Project of Jiangsu Province (No. BE2022811). Furthermore, the work was also supported by the Big Data Computing Center of Southeast University.
Authors' Contact Information: Shipeng Zhu, School of Computer Science and Engineering, Southeast University, Nanjing, China and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, Nanjing, China; e-mail: shipengzhu@seu.edu.cn; Jun Fang, School of Computer Science and Engineering, Southeast University, Nanjing, China and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, Nanjing, China; e-mail: 220212049@seu.edu.cn; Pengfei Fang, School of Computer Science and Engineering, Southeast University, Nanjing, China and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, Nanjing, China; e-mail: fangpengfei@seu.edu.cn; Hui Xue (corresponding author), School of Computer Science and Engineering, Southeast University, Nanjing, China and Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, Nanjing, China; e-mail: hxue@seu.edu.cn.

ACM 1551-6865/2024/11-ART379
https://doi.org/10.1145/3696209

1 Introduction

Perceiving text efficiently in scene images is crucial for various real-world applications, including social media analysis [52], robot navigation [28], intelligent transportation [13], and privacy protection [36]. Central to this goal, numerous studies within the community have delved into text recognition [46], detection [8], and spotting [51]. Distinct from the above scenarios, retrieving the specific text from a vast collection of natural images is an interesting problem, which can be encapsulated by the paradigm of **scene text (ST)** retrieval. Introduced in [21], ST retrieval involves localizing and identifying ST instances in images based on a given query text (shown in Figure 1(a)). Essentially, this task aims to develop a content analysis system typical of multi-media applications, which establishes the relevance between query text and specific regions within ST images. Such a setting is pivotal for our daily life. For example, the act of finding images with specific textual content from an expansive gallery of smartphone photographs, or leveraging partial recollected text to discover more comprehensive textual content, is a common yet extremely time-consuming task. As a result, the functionality for ST image retrieval has been incorporated into the photo album features of some mobile systems. Furthermore, this task can serve as a critical component in more applications, e.g., text-aware event re-identification [24], automatic navigation [42], and key information extraction [34], shown in Figure 1(b). However, ST retrieval presents two main challenges: (1) Locating instances of ST graphics that match the query terms. ST instances within images are typically dense, varied, and of irregular sizes, which can lead to ambiguities for the model; (2) Achieving a balance between retrieval performance and processing speed, as numerous retrieval tasks are typically subject to real-time demands. Notably, common image-text retrieval systems [7, 30, 50], although showing good performance in general multi-media scenarios, fall short of resolving this problem. Specifically, these methods concentrate on the alignment of **text-image (T&I)** semantics, resulting in an inability to handle the complex aggregations of multiple characters [56, 59].

Over the years, several approaches [17, 19] have been developed to tackle the ST retrieval problem, with many relying on text spotting methods. As illustrated in Figure 2(a), these methods aim to detect and recognize all text instances within scene images to solve the first challenge. Subsequently, the recognized ST instances are matched with the query text. Consequently, they can achieve effective retrieval performance for text instances that are of substantial size and high quality. However, a notable issue of retrieval omissions arises from this **"Text-Text" (T&T)** paradigm. Specifically, while striving for both detection and recognition accuracy, these methods might overlook vague text instances. As a result, spotting methods might not generate adequate candidate proposals, leading to ignorance of tiny and sub-word instances [38]. Meanwhile, within this paradigm, the majority fail to effectively balance the precision of text recognition with processing speed [42], leading to limitations in many applications. In contrast, some tailored methods, as shown in Figure 2(b) and referenced in [9, 21, 38], approach the problem as a direct T&I cross-modality matching paradigm. The main strategy involves locating potential ST instance proposals and subsequently projecting them, along with the query text, into a shared embedding space for similarity evaluation. This paradigm can initially select a wide array of potential text areas. Employing subsequent similarity evaluation, it advances the selection process, significantly alleviating the shortcomings of omission in T&T methodologies. A significant drawback, however, is that these methods often neglect the modality gap between textual and visual data [29], thereby constraining their effectiveness. Most recently, Wen et al. introduced a visual matching paradigm [44]. In this **"Image-Image" (I&I)** approach, the query text is converted into its corresponding graphical image, as seen in Figure 2(c). The visual similarity between this representation and ST instance proposals is then assessed. This technique seeks to resolve the modality gap by aligning visual data. Yet it is constrained by its heavy reliance on visual features, resulting in challenges when distinguishing between



(a) Paradigm of scene text retrieval (b) Application examples of scene text retrieval

Fig. 1. (a) Given a query text "HOTEL," the ST retrieval method aims to search all images containing "HOTEL" from the gallery and output their locations in the images. (b) Downstream application examples of ST retrieval.

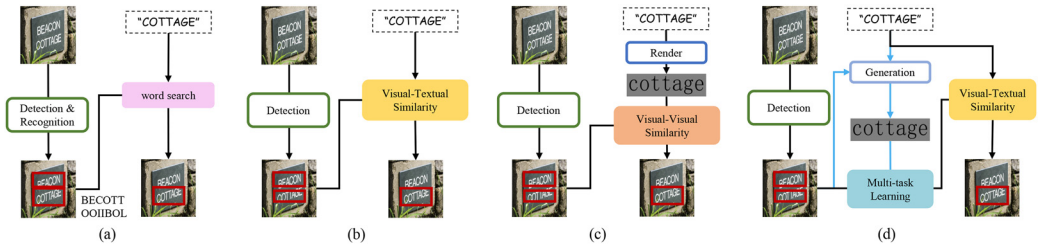


Fig. 2. The illustration of different kinds of ST retrieval methods. (a) Text spotting methods localize and recognize all the text instances used for T&T retrieval. (b) Conventional retrieval methods consider the task as the T&I cross-modal matching task. (c) Wen et al. convert query words into images to retrieval text instances in an I&I manner. (d) Our method stylized middle modality learning (SM^{2}L) proposes the "Text-Image-Image" (T&I&I) paradigm during the training phase (denoted by the blue lines). The middle modality is omitted during the inference phase.

analogous characters, such as "c" and "e" or "a" and "0," particularly in the context of low-quality image regions.

At its core, ST retrieval seeks to identify specific ST instances that align with the content of a given query text. However, as aforementioned, previous approaches struggle to achieve both semantic coherence across modalities and visual alignment between the query text and the ST instances. Such shortcomings can lead to problems of omissions and erroneous associations in the retrieval process, which in turn, restrict the effectiveness of the retrieval capabilities. These analyses inspire us to approach the challenge from a dual perspective. Specifically, we aim to maintain semantic consistency while introducing visual constraints during the optimization phase. A stylized middle modality is designed to retain the content of the query text while adopting the style of the ST instances. As a result, we introduce a novel **stylized middle modality learning (SM^{2}L)** framework, illustrated in Figure 2(d). The SM^{2}L framework encompasses three distinct branches: the ST image branch, the **Stylized Graphical Text (SGT)** image branch, and the **Text (T)** branch. During training, the ST branch generates ST instance proposals from candidate scene images. Simultaneously, the SGT branch, serving as a middle modality, transforms the query text into SGT. This transformation leverages the visual attributes of both the query text and the ST instance proposals, e.g., character structure and texture. The T branch, on the other hand, is devised

to enforce semantic consistency. In light of [57], we deploy a multi-task learning approach for the three modalities, encompassing ranking learning, character learning, and adversarial learning. This strategy jointly optimizes the visual and semantic consistency across visual and textual modal features. Therefore, SM^{2}L features three technique innovations, along with the superior performance they bring: (1) We introduce, for the first time in the domain of ST retrieval, a triple-modality training framework. This framework employs a stylized middle modality to act as a conduit, connecting the visual information of the ST to the textual attributes of the query terms, which markedly mitigates the challenges posed by the modality gap. Furthermore, this framework ensures efficiency by maintaining only the dual-modality similarity calculations during the inference phase; (2) Our approach employs simple yet effective networks to construct the triple branches. Here, the SGT modality can capture the visual correlation between the query text and the ST instances, while the T modality addresses the ambiguity arising from visually similar words through leveraging semantic understanding. (3) The multi-task learning paradigm incorporates multi-granularity predictions and the multi-alignment of feature similarity. In practice, these strategies significantly enhance training efficiency.

The contributions of this article are shown as follows:

—We introduce a novel SM^{2}L framework, a pioneering approach that incorporates a stylized visual middle modality. This modality acts as a bridge, connecting the query text with the ST image, thus effectively addressing the challenges in ST retrieval.
—The SM^{2}L framework uniquely leverages a multi-task learning paradigm, integrating the ST image, SGT image, and textual data. This paradigm reconciles the semantic and visual consistency of the query text and ST instances during the optimization phase.
—Extensive experiments demonstrate that the proposed SM^{2}L achieves **state-of-the-art (SOTA)** performance on three benchmark datasets. Ablation studies verify the effectiveness and essential contributions of the various components in the SM^{2}L framework.

2 Related Work

2.1 ST Spotting

ST retrieval inherently demands two key capabilities: searching for images that contain the queried text and pinpointing the specific text instance. Image-level and Optical Character Recognition-based retrieval methods fall short when faced with typical ST images, which often feature dispersed text instances and intricate backgrounds. A natural solution to ST retrieval involves spotting the text, i.e., pinpointing and identifying the texts within images and then retrieving them in a "T&T" fashion. A landmark in spotting work, named Mask TextSpotter [19], utilizes an end-to-end framework for text of varying shapes and has displayed commendable results. Several subsequent endeavors employ different foundational structures to elevate detection and recognition precision. For instance, the ABCNet series [17, 18] introduce the adaptive Bezier curve network, a strategy that proves effective in managing oriented and curved ST. [6] integrates a language model in character decoding, notably improving recognition precision. Following the Transformer [5] style process, TESTR [55] casts the DETR [3] framework to decode the locations and characters of text instances in parallel. Furthermore, the SOTA method, ESTextSpotter [14], uniquely employs a single decoder to model discriminative and interactive features, allowing it to detect and recognize text in one seamless action. Other methods exploit simplified features to improve the spotting speed. PAN++ [41] leverages the lightweight module to detect text kernel instead of complex segmentation. By imposing point gathering Connectionist Temporal Classification loss [11], PGNet [40] excavates the pixel-level character feature map, avoiding time-consuming operations like Non-Maximum Suppression [54]. Although such spotting methods have promising performance on text spotting,

they are always prone to miss some potential text instances due to different optimization criteria with the retrieval task.

2.2 ST Retrieval

Cross-media retrieval [45] has long been a hot topic within the community. In most cases, the primary objective typically revolves around the retrieval of particular images leveraging textual information. Some advances [47, 48, 58], employing mechanisms like hash, achieve remarkable results in general scenarios. However, as previously mentioned, these methods encounter bottlenecks when faced with various text instances in scene images, rendering them inapplicable to ST retrieval.

Conventional tailored text retrieval methods [1, 2, 33] primarily target cropped document text images, gauging the distance between images and query terms via neural networks. Nonetheless, these methods are limited in real-life scenarios, where arbitrarily shaped STs are blended with complex backgrounds. IRTC [21] first introduces the ST retrieval task and provides a character-centric approach. Seeking to bypass efficiency issues inherent in the two-stage paradigm, subsequent works like [9] design an end-to-end trainable network, drawing inspiration from YOLO [26]. Their method involves ranking the similarity between the PHOCs [2] of query text and detected text instance proposals. Another innovative approach, RL-STR [20], poses the task as an efficient sequential selection from the set of extremal regions, leading to real-time ST retrieval performance. Recent advancements have seen methods, such as the one introduced in [38], that directly calculate the cross-modal cosine similarity between text image instance proposals and query text. The results from these methods have showcased marked improvements in retrieval performance. However, these "T&I" methods, despite their innovations, have their limitations. Some are merely adaptations of existing "T&T" methods, while others grapple with bridging the vision-language modality gap. A study [44] offers a fresh paradigm in this context. This approach aims to transform the cross-modality retrieval challenge, viewing it as a problem rooted in similarity measurement within the visual modality. However, the "I&I" approach predominantly emphasizes visual similarity, inadvertently overlooking the textual ambiguities that arise from visually similar characters.

3 Methodology

This section details the proposed ST retrieval machine in a top-down fashion: Initially, we provide an overview of the method, followed by a detailed explanation of the network architecture. Subsequently, the learning tasks employed for optimization are introduced.

3.1 Overview

We commence with an outline of the proposed SM^{2}L framework. The network architecture is depicted in Figure 3. In the training phase, the framework incorporates a ST image branch, a SGT image branch, and a T branch. The ST branch extracts features from all potential ST instance proposals, represented as F^{st}, for a given ST image (detailed in Section 3.2). For a specific query text Q, the SGT branch converts its graphical representation R into a stylized graphical word SG. This transformation ensures alignment with the predefined content of R and style criteria SG, producing a visual feature F^{sgt} (elaborated in Section 3.3). Concurrently, the T branch generates a textual feature F^{t} for the query text, as explained in Section 3.4. These three features facilitate the optimization of the network within a multi-task learning framework, discussed in Section 3.5. During inference, the system employs only the ST and T branches, calculating the cosine similarity between these two types of features for the ranking task. This approach preserves the processing speed compared to existing SOTA methods, a claim substantiated in Section 4.4.

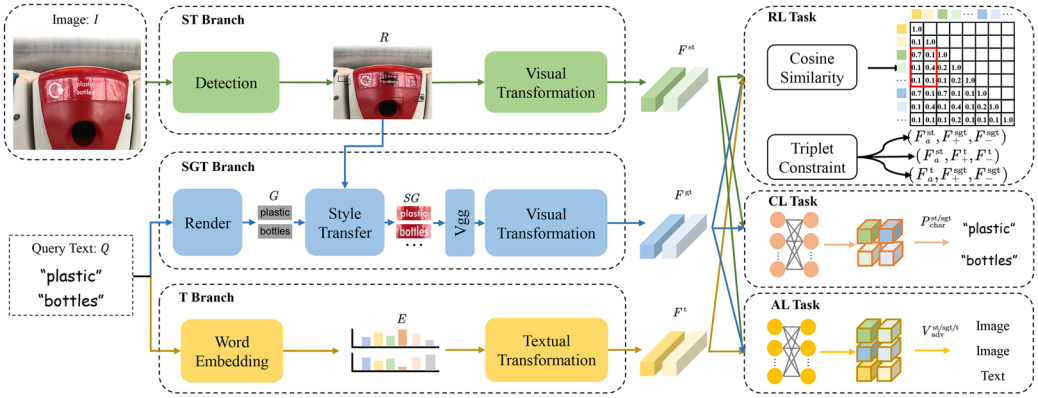


Fig. 3. The overview architecture of the proposed SM^{2}L. "RL," "CL," and "AL" denote ranking learning, character learning, and adversarial learning, respectively.

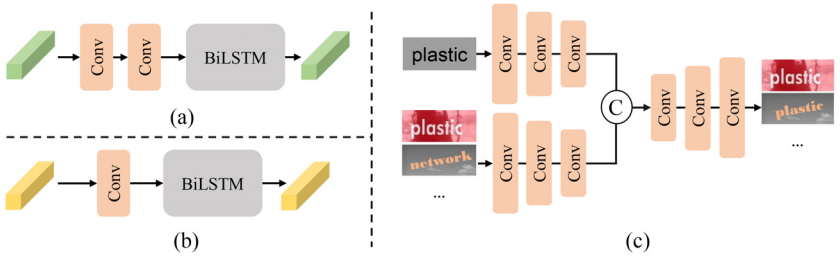


Fig. 4. The architecture of different modules. (a) The visual transformation module with two convolution layers and one BiLSTM layer. (b) The textual transformation module with one convolution layer and one BiLSTM layer. (c) The style transfer module with two encoders and one decoder. "C" denotes concatenation operation.

3.2 ST Branch

The ST branch is designed to detect all potential ST instances and extract individual visual features for each. To achieve this, we utilize a general anchor-free object detector, designated as Det^{st}, for the generation of RoI features. This process is expressed as $R = \text{Det}^{\text{st}}(I), R = r_i{}^N i = 1 \in \mathbb{R}^{N\times CR\times W_R\times H_R}$. Subsequently, a visual transformation module, Trans^{st}, is employed to derive discriminative features from ST, which are crucial for subsequent cross-modal calculations. As depicted in Figure 4(a), this module comprises two convolution layers followed by a BiLSTM layer. Previous studies such as [31, 42] have demonstrated the effectiveness of this configuration in various ST-related tasks. The feature extraction is formalized as $F^{\text{st}} = \text{Trans}^{\text{st}}(R) \in \mathbb{R}^{N\times C\times W}$, where W and C denote the width and channel dimensions of the features, respectively.[1]

3.3 SGT Branch

One of the primary challenges in existing cross-modal retrieval methods is the significant modality gap between candidate images and query text, which adversely affects retrieval performance. To cope with this issue, we employ an SGT branch to produce a stylized middle modality feature of the query text and the ST instance proposals, thereby bridging the gap between the two modalities.

[1]Post-convolution, the feature height is reduced to 1, which is then squeezed to facilitate processing by the BiLSTM layer.

As shown in the SGT branch of Figure 3, we first utilize a widely used rendering generator [12] to produce image-format data for the query text, as $G = \text{Gen}^{\text{sgt}}(Q)$, where $G \in \mathbb{R}^{1\times W_Q\times H_Q}$. Subsequently, a compact encoder-decoder style transfer module, shown in Figure 4(c), processes G and R to produce stylized text images SG. Notably, each query text can generate N SGTs, corresponding to the number of ST instance proposals, which can be formatted as:

$$SG = \text{StyTrans}^{\text{sgt}}(Q, R), SG = \{sg_i\}_{i=1}^{N} \in \mathbb{R}^{N\times C_R\times W_Q\times H_Q}. \tag{1}$$

Subsequently, a feature extraction module, the tiny version of Vgg [32], extracts the visual information from SG. Then, a visual transformation module is applied to understand the **graphical text (GT)** feature. We summarize this process as: $F^{\text{sgt}} = \text{Trans}^{\text{sgt}}(\text{Vgg}(SG)) \in \mathbb{R}^{(M*N)\times C\times W}$. Notably, the text number M is the same as the N of F^{st} in the training phase.

3.4 T Branch

Intuitively, following the I&I paradigm, the ST and SGT branches can be directly used for the retrieval task. However, this approach may overlook the semantic features, potentially leading to retrieval errors when visually similar texts are present. For instance, words such as "same" and "some" might appear close in visual feature space due to their structural similarities, thus skewing the similarity calculation in retrieval tasks. Moreover, the style transfer module in the SGT branch could impede inference speed. To address these issues, we incorporate the T branch in the training phase to anchor semantic similarity, while omitting the SGT branch during inference for efficiency.

In the T branch, following the setting in [38], we transform the query text Q into a semantic feature space. Specifically, Q undergoes processing through an embedding layer and bilinear interpolation to produce the word embedding $E \in \mathbb{R}^{N\times C_E\times W_E}$. Subsequently, as depicted in Figure 4(b), the textual transformation module maps E to the feature F^{t}, defined as: $F^{\text{t}} = \text{Trans}^{\text{t}}(E) \in \mathbb{R}^{N\times C\times W}$.

3.5 Multi-Task Learning

As discussed from Sections 3.2 to 3.4, the proposed SM^{2}L framework can offer three types of features, namely, $F^{\text{st}}, F^{\text{sgt}}, F^{\text{t}}$. This encourages us to devise a multi-task learning approach to constrain the visual feature coherence and text semantic feature consistency in the training phase, thereby improving the retrieval performance of the machine. In this context, we use three tasks in the learning procedure, namely, ranking learning, character learning, and adversarial learning.

3.5.1 Ranking Learning Task. We first employ the ranking learning task, constrained by the cosine similarity loss and triplet loss, to learn a joint embedding space for the visual and semantic features. As shown in Figure 3, cosine similarity between the above three features is applied for ranking learning. In light of [38], we calculate the pairwise similarity between any two features, such that the visual and text features can be aligned. Before presenting the loss, we denote the features from any two modalities as $F^p, F^q \in \mathcal{F} = \{F^{\text{st}}, F^{\text{sgt}}, F^{\text{t}}\}$. In this context, an encoder, denoted by Enc, first encodes any two modalities of features, F_i^p and F_j^q, into vector representations. Then, the cosine similarity of those two vectors can be calculated as:

$$S_{i,j}(F^p, F^q) = \frac{\tanh\left(\text{Enc}\left(F_i^p\right)\right)^\top \tanh\left(\text{Enc}\left(F_j^q\right)\right)}{||\tanh\left(\text{Enc}\left(F_i^p\right)\right)|| \cdot ||\tanh\left(\text{Enc}\left(F_j^q\right)\right)||}, \tag{2}$$

where $i, j \in [1, N]$. Notably, only the $S_{i,j}(F^{\text{st}}, F^{\text{t}})$ (illustrated in the red box of Figure 3) is performed as the ranking basis during the inference phase. Since there is no ground-true label for the cosine similarity, we employ another similarity metric as the supervision signal. Specifically, given the

label text $(l_{F_i^p}, l_{F_j^q})$ of (F_i^p, F_j^q), the label of similarity can be denoted as:

$$\hat{S}_{i,j}(F^p, F^q) = 1 - \frac{\mathrm{ED}\left(l_{F_i^p}, l_{F_j^q}\right)}{\max\left(|l_{F_i^p}|, |l_{F_j^q}|\right)}, \tag{3}$$

where ED denotes the edit distance [15]. Therefore, the loss function is shown as follows:

$$\left(\mathcal{L}_{\mathrm{s}}\left(F^p, F^q\right)\right)_{i,j} = \mathrm{SL}\left(S_{i,j}(F^p, F^q), \hat{S}_{i,j}(F^p, F^q)\right), \tag{4}$$

$$\mathcal{L}_{\cos} = \sum_{(F^p, F^q)\in\mathcal{F}} \frac{1}{N} \sum_{i}^{N} \max_{j}^{N} \left(\mathcal{L}_{\mathrm{s}}\left(F^p, F^q\right)\right)_{i,j}, \tag{5}$$

where SL denotes the smooth L1 loss.

Meanwhile, triplet loss [4, 27], popularly used for retrieval tasks, is also used to learn a discriminative embedding space. In our task, we elaborate on it as a cross-modality triplet loss. Given the anchor feature F_a^p, the positive feature F_+^q, and the negative feature F_-^q, $\mathcal{L}_{\mathrm{trip}}$ can be shown as:

$$\begin{aligned}\mathcal{L}_{\mathrm{pair}}(F_a^p, F_+^q, F_-^q) = &\max\left(0, \mu - |S\left(F_a^p, F_+^q\right)|_m\right. \\ &\left. + |S\left(F_a^p, F_-^q\right)|_m\right),\end{aligned} \tag{6}$$

$$\begin{aligned}\mathcal{L}_{\mathrm{trip}} = &\mathcal{L}_{\mathrm{pair}}(F_a^{\mathrm{st}}, F_+^{\mathrm{sgt}}, F_-^{\mathrm{sgt}}) + \mathcal{L}_{\mathrm{pair}}(F_a^{\mathrm{st}}, F_+^{\mathrm{t}}, F_-^{\mathrm{t}}) \\ &+ \mathcal{L}_{\mathrm{pair}}(F_a^{\mathrm{t}}, F_+^{\mathrm{sgt}}, F_-^{\mathrm{sgt}}),\end{aligned} \tag{7}$$

where μ denotes the margin and $|\cdot|_m$ denotes the mean absolute value for the similarity. Notably, in our triplet loss, the anchor sample and positive/negative one come from different modality data. In summary, the ranking loss can be formulated as:

$$\mathcal{L}_{\mathrm{rank}} = \mathcal{L}_{\cos} + \mathcal{L}_{\mathrm{trip}}. \tag{8}$$

3.5.2 Character Learning Task. Although the ranking learning task optimizes the similarity of features, excavating the explicit characters of visual modalities can also improve semantic discrimination in multi-task learning. Specifically, we utilize an **Multi-Layer Perceptron (MLP)**-based text classifier to predict the text strings of images in ST and SGT modalities. As shown in Figure 3, the predicted text $P_{\mathrm{char}} \in \mathcal{P}_{\mathrm{char}} = \{P_{\mathrm{char}}^{\mathrm{st}}, P_{\mathrm{char}}^{\mathrm{sgt}}\}$ is first labeled by its ground-truth text, denoted by $\hat{P}_{\mathrm{char}}$. Then, the cross-entropy loss $\mathcal{L}_{\mathrm{ce}}$ is adapted to constrain each character of the text string, defined as:

$$\mathcal{L}_{\mathrm{char}} = \mathcal{L}_{\mathrm{ce}}(P_{\mathrm{char}}^{\mathrm{st}}, \hat{P}_{\mathrm{char}}^{\mathrm{st}}) + \mathcal{L}_{\mathrm{ce}}(P_{\mathrm{char}}^{\mathrm{sgt}}, \hat{P}_{\mathrm{char}}^{\mathrm{sgt}}). \tag{9}$$

where $(P_{\mathrm{char}}, \hat{P}_{\mathrm{char}}) \in \mathbb{R}^{N\times C_P\times W_P}$ and (C_P, W_P) denotes the number of potential characters and the max length of the predicted text, respectively.

3.5.3 Adversarial Learning Task. To further minimize the modality gap, we employ adversarial learning in the training phase. That is, the min-max optimization process maps the features from all three modalities to the same distribution. Specifically, an MLP-based modality classifier, acting as the discriminator in Generative Adversarial Network [10], receives each feature as input and produces the corresponding modality probability $P_{\mathrm{adv}} \in \mathcal{P}_{\mathrm{adv}} = \{P_{\mathrm{adv}}^{\mathrm{st}}, P_{\mathrm{adv}}^{\mathrm{sgt}}, P_{\mathrm{adv}}^{\mathrm{t}}\}$. The discriminator can predict whether the feature is derived from image or text format data. The adversarial loss is given by:

$$\begin{aligned}\mathcal{L}_{\mathrm{adv}} = &\mathcal{L}_{\mathrm{ce}}(P_{\mathrm{adv}}^{\mathrm{st}}, \hat{P}_{\mathrm{adv}}^{\mathrm{st}}) + \mathcal{L}_{\mathrm{ce}}(P_{\mathrm{adv}}^{\mathrm{sgt}}, \hat{P}_{\mathrm{adv}}^{\mathrm{sgt}}) \\ &+ \mathcal{L}_{\mathrm{ce}}(P_{\mathrm{adv}}^{\mathrm{t}}, \hat{P}_{\mathrm{adv}}^{\mathrm{t}}),\end{aligned} \tag{10}$$

where $\hat{P}_{\mathrm{adv}} \in \mathbb{R}^N$ is the ground-truth modality label for each feature, i.e., image or text.

3.6 Training Objective

In the training phase, the proposed network is optimized by a multi-task learning loss, defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{dec}} + \mathcal{L}_{\text{sty}} + \mathcal{L}_{\text{rank}} + \mathcal{L}_{\text{char}} - \mathcal{L}_{\text{adv}}, \tag{11}$$

where $\mathcal{L}_{\text{dec}}$ and $\mathcal{L}_{\text{sty}}$ denote the detection loss [49] and the style transfer loss [25, 43], respectively. Notably, we follow the adversarial network [10] to optimize the parameters, i.e., the parameters of the discriminator and the three branches are alternately optimized.

4 Experiments

In this section, we first introduce the evaluation metric, i.e., **mean Average Precision (mAP)**. Then, the datasets and implementation details are described. Lastly, we perform comparison experiments and ablation studies to validate the advantages of the proposed SM^{2}L.

4.1 Evaluation Metrics

In the evaluation process, we routinely utilize mAP and **Frames Per Second (FPS)** to assess retrieval performance and inference speed, respectively. The computation of mAP is grounded in the cosine similarity between the query text and text proposals during the testing phase. Based on the similarity scores, the retrieved instances are sorted in descending order, along with their corresponding ground truths. Given $\mathcal{N}$ retrieved instances, the precision and recall values are calculated in the condition of predicting correctly the top k instances separately, where $1 \leqslant k \leqslant \mathcal{N}$. These calculations contribute to a Precision-Recall curve, with the **Average Precision (AP)** being the area under this curve. The mAP is then derived as follows:

$$\text{mAP} = \frac{1}{\mathcal{K}} \sum_{i=0}^{\mathcal{K}} \text{AP}_i, \tag{12}$$

where $\mathcal{K}$ stands for the number of query texts, and AP_i denotes the AP corresponding to each query text.

4.2 Datasets

Following the common setting of existing methods, we employ two datasets for training and three other ones for testing, as shown below:

Training Datasets. SynthText-900k [9]:[2] it contains 900k synthetic ST images generated by the rendering model [12], which is widely used in the area of ST tasks. ***Multi-lingual Scene Text 5k (MLT-5k)***:[3] a subset of MLT [22], which contains 5,000 English language images.

Testing Datasets. ***Street View Text (SVT)*** [39]:[4] it contains 349 images gathered from Google Street View. Each image is annotated with bounding boxes and word labels. ***IIIT Scene Text Retrieval (STR)*** [21]:[5] it contains 10,000 images and 50 query words collected from Google and Flickr. Notably, the annotations are merely word labels without bounding boxes. ***Coco Text Retrieval (CTR)***:[6] since [38] does not offer a specific dataset, we select 1,223 images with complex text instances by 50 query words from Coco-Text [37], which are annotated by word labels and bounding boxes.

[2] http://datasets.cvc.uab.es/rrc/SynthText_90KDict.tar
[3] https://github.com/lanfeng4659/STR-TDSL
[4] https://tc11.cvc.uab.es/datasets/SVT_1
[5] https://cvit.iiit.ac.in/research/projects/cvit-projects/the-iiit-scene-text-retrieval-str-dataset
[6] https://bgshih.github.io/cocotext/

4.3 Implementation Details

We implement our model using Pytorch [23] with four NVIDIA RTX 3090 GPUs for training and only one GPU for testing. To enhance feature extraction, we introduce a multi-scale variant of the SM^{2}L framework, incorporating the multi-scale operation in the detection module, i.e., **Adaptive Training Sample Selection (ATSS)** [53]. In the ST branch, the RoI features R from the detection module are described by the dimensions C_R, W_R, H_R, representing channel, width, and height, respectively. Similarly, in the GT branch, W_Q, H_Q refer to the width and height of the text image G and SG, respectively. In the T branch, C_E, W_E correspond to the channel and width of the word embedding E. The size of each text image proposal r_i in R and word embedding E are $256 \times 4 \times 15$ and 256×15, respectively. The channel C and width W of features $F^{\mathrm{st/sgt/t}}$ are both set to 128 and 15. The C_P and W_P of P_{char} and $\hat{P}_{\mathrm{char}}$ are set to 37 and 15 in the character learning task. The size of the generated GT image G and SG is $1 \times 32 \times 128$ in both the training and inference phases. We follow the rendering model [12] and adopt Arial as the font of G, to ensure the clarity and legibility of characters.

In aligning with the basic settings and word augmentation strategies outlined in [38], our training phase is divided into two distinct stages, both utilizing SGD optimizers with a weight decay of 0.0001 and a momentum of 0.9. Initially, in the pre-training stage, the model undergoes training on the Synth-900k dataset for 112,500 iterations, starting with a learning rate of 0.01, which is reduced by a factor of 0.1 every 37,500 iterations. This stage uses a batch size of 64, and images are resized to 640×640. During the fine-tuning stage, the MLT-5k dataset with data augmentation strategies is employed, training the model with a batch size of 32 for 10,000 iterations. The initial learning rate is 0.001, decreasing to 0.0001 after 50,000 iterations. Our empirical observations indicate that the total loss function shows negligible sensitivity to the weights of sub-loss functions, prompting us to set all weights to 1. Furthermore, the margin value μ in $\mathcal{L}_{\mathrm{trip}}$ is fixed at 0.1. Notably, all three branches undergo simultaneous training, and only the ST branch and T branch are employed for inference. For inference, we resize the image width to 1,150 while preserving the original aspect ratio.

4.4 Comparison with SOTA Methods

We compare our SM^{2}L with a range of leading spotting and retrieval methods on the SVT, STR, and CTR datasets. For the spotting methods, we include high-performance spotters such as Mask-TextSpotter V3 [16], ABCNet V2 [18], ABINet++ [6], TESTR [55], ESTextSpotter [14], as well as real-time spotters like PGNet [40] and PAN++ [41]. The latter retrieval methods consist of several representative tailored ones, such as IRTC [21], YOLO-STR [9], RL-STR [20], TD-STR [38], and VM-STR [44]. To evaluate the retrieval performance of spotting methods, we utilized the normalized edit distance metric to determine the similarity score between recognized texts and their corresponding labels. All experimental results of the above methods are derived from the officially released models.

4.4.1 Quantitative Analysis. To delve into the specific performance metrics, we report the quantitative results in Table 1.

Our SM^{2}L model, especially in its multi-scale implementation, SM^{2}L(MS), not only matches the retrieval speed of the TD-STR series but also demonstrates SOTA performance. Notably, this method outperforms the T&T method ESTextSpotter [14] and the T&I method TD-STR(MS) [38] by 6.82 and 1.90, respectively. Our vanilla solution (avg. 82.66) exceeds the performance of leading fast methods like PGNet [40] with improvements of 9.29 in average mAP across three datasets, nearing the 83.65 of TD-STR(MS). It is noteworthy that conventional T&T methods show limited performance on the CTR dataset, which often contains images with small text instances, highlighting the inherent

Table 1. Performance Comparison (mAP Score) on SVT, STR, and CTR

Type	Model	Venue	SVT	STR	CTR	AVG	FPS
T&T	PAN++ [41]	TPAMI'21	80.29	69.54	69.85	73.23	13.44
	PGNet [40]	AAAI'21	80.78	74.57	64.75	73.37	**33.33**
	MaskSpotter V3 [16]	ECCV'20	83.02	72.96	64.32	73.43	2.03
	ABCNet V2 [18]	TPAMI'22	86.12	78.18	63.19	75.83	4.17
	ABINet++ [6]	TPAMI'23	85.50	78.70	63.32	75.84	4.00
	TESTR [55]	CVPR'22	86.37	79.16	68.44	77.99	3.45
	ESTextSpotter [25]	ICCV'23	85.94	78.39	71.85	78.73	5.00
T&I	IRTC [21][a]	ICCV'13	56.24	42.70	-	-	-
	YOLO-STR [9]	ECCV'18	84.99	69.55	41.07	69.38	11.11
	YOLO-STR(MS) [9]	ECCV'18	86.32	71.92	42.79	71.32	3.57
	RL-STR[a] [20]	PRJ'21	85.74	71.67	-	-	-
	TD-STR [38]	CVPR'21	89.24	76.94	73.59	79.36	18.01
	TD-STR(MS) [38]	CVPR'21	91.57	81.15	78.23	83.65	3.66
I&I	VM-STR[a] [44]	WSDM'23	90.95	77.40	-	-	-
T&I&I	SM^2L (ours)	/	92.05	79.32	76.60	82.66	18.01
	SM^2L(MS) (ours)	/	**93.43**	**83.00**	**80.21**	**85.55**	3.66

[a]indicates the method does not release code, and some results, e.g., CTR, AVG and FPS, cannot be obtained which are replaced by None ("-"). AVG stands for the average mAP score among the three datasets. "MS" means multi-scale feature extraction in the detection module during inference. The best scores are bold.

limitations of this paradigm where detection capability is constrained by the recognition sub-task. In contrast, our method exhibits robust generalization across all three datasets.

4.4.2 Qualitative Analysis. In addition to the quantitative analysis, we present qualitative experiments to further demonstrate the superiority of our method. First, we conduct a comparative analysis of SM^2L with all the open source methods previously mentioned. Figure 5 showcases the localization accuracy of these methods using various query texts and their corresponding images. Notably, SM^2L exhibits superior localization precision. For instance, in the third row of Figure 5, SM^2L accurately delineates the target text, whereas competing methods erroneously identify semantically similar words. This underscores the effectiveness of the designed stylized middle modality. Additionally, as shown in the second row, SM^2L correctly localizes "pak," distinguishing it from the similar-looking "PAYA," highlighting the efficacy of semantic alignment in the multi-task learning of our approach. Conversely, conventional spotting methods, which pre-identify and recognize all the text instances in a scene image, falter with ambiguous queries. An example is their struggle with "market" when "supermarket" is present, as depicted in the first row. This limitation reduces their utility in the ST retrieval task. To investigate the flexible retrieval capability of different paradigms, we present the retrieval results of different approaches on one image, each based on different query texts. As illustrated in Figure 6, our proposed SM^2L properly localizes the target text proposals according to different query texts. In contrast, T&T methods commonly struggle to process sub-words effectively, such as "yart" and "city." Furthermore, the T&I method, TD-STR, is misled by similar or completely irrelevant areas, resulting in erroneous retrieval outcomes.

Following common practices [38], we first visually demonstrate the retrieval efficacy of our SM^2L model on the CTR dataset. This is depicted in Figure 7, where we showcase the top-3 retrieval results for each query text. For example, our proposed method effectively locates the correct "restaurant" among redundant blurry texts in complex backgrounds, further reinforcing its robustness in real-life

Fig. 5. The retrieval results in SVT. The query texts are "market," "pak," "crest," "hut," "fitness," "zone," "hotel," "joe," and "town" from left to right. The comparison method "MTS v3" denotes Mask TextSpotter v3.

Fig. 6. The retrieval results on an image based on query texts "arts," "city," "cityarts," and "yart." The comparison method "MTS v3" denotes Mask TextSpotter v3.

scenarios. Driven by these observations, SM^{2}L demonstrates SOTA performance both quantitatively and qualitatively.

4.5 Ablation Study

In this section, we provide ablation studies to verify the effectiveness of the **Text-Image-Image (T&I&I)** paradigm, the style transfer module, the multi-task learning scheme, and the detection module. All the experiments are studied on three datasets, i.e., SVT, STR, and CTR.

4.5.1 Effect of the T&I&I Paradigm. In this work, we craft a novel T&I&I paradigm to construct the SM^{2}L network, achieving outstanding performance in ST retrieval tasks. To highlight the significance of each modality in the model, we carry out comparative experiments involving various branch combinations. A notable addition to our experimental setup is the introduction of a simplified GT branch. This branch, different from the SGT branch, excludes the style transfer module, serving as a middle modality to assess the effectiveness of our advanced SGT branch. Additionally, we tailored the learning objectives for each branch combination as needed. The results,

Fig. 7. The top-3 retrieval results of the proposed SM^{2}L based on query texts "adidas," "airlines," "restaurant," and "donuts" in CTR.

Table 2. Ablation Study of the T&I&I Paradigm

	Branch				SVT	STR	CTR	AVG	FPS
	ST	GT	SGT	T					
(i)	✓	-	-	✓	91.07	78.04	76.29	81.80	18.01
(ii)	✓	✓	-	-	90.10	78.08	75.56	81.25	**18.46**
(iii)	✓	✓	-	✓	91.63	78.13	76.32	82.02	18.01
(iv)	✓	-	✓	✓	**92.05**	**79.32**	**76.60**	**82.66**	18.01

"✓" stands for utilizing the branch in the training phase. The gray cell denoted the branch used in the inference phase. The bold number denotes the best performance.

detailed in Table 2, lead to several key conclusions: (1) The full SM^{2}L model with all three branches during training surpasses the performance of both T&I and I&I paradigms (see (i), (ii), and (v)). This validates our approach of concurrently maintaining semantic and visual coherence between the modalities. (2) The inclusion of our SGT branch yields a substantial performance improvement (+0.64 in avg. map) compared to the normal GT (see (iii) and (v)). This confirms the efficacy of the style transfer module in bridging the ST and T branches. (3) Within the T&I&I paradigm, the retrieval of ST instances using stylized graphic text is as effective as using direct textual data (see (iv) and (v)). This demonstrates the capability of our model to align different modalities of the same content. Nevertheless, the approach that utilizes textual data for retrieval exhibits greater computational efficiency, thus establishing an optimal balance between accuracy and efficiency.

4.5.2 Effect of the Style Transfer Module. In this study, experiments were conducted to assess the contribution within the style transfer module, a critical component within the SGT branch of the SM^{2}L network. As depicted in Figure 4 in Section 3.3, this module, adopting a streamlined encoder-decoder architecture, aims to transfer the style from ST instance proposals to GT. For comparative analysis, we implemented a variation of SM^{2}L, integrating MOSTEL [25], a conventional ST editing method, to perform style transference. However, as indicated in Table 3, the incorporation of the pre-trained MOSTEL model resulted in decreased retrieval accuracy. This is attributed to the poor adaptation of MOSTEL to ST instances with various properties, such as small zones and low

Table 3. Ablation Study of Style Transfer Module

Module	SVT	STR	CTR	AVG	Param.	FLOPs
MOSTEL	90.41	77.02	73.93	80.45	59.78M	18.01 G
Ours	**92.05**	**79.32**	**76.60**	**82.66**	**4.96M**	**2.29 G**

MOSTEL [25] is a typical scene text editing method. Here, we adopt a frozen model with pre-trained parameters in the training phase. The "Param." and "FLOPs" denote the parameters and computational complexity of SM^{2}L with different modules during the training phase, respectively. The bold number denotes the best performance.

Table 4. Ablation Study of Multi-Task Learning

Loss	SVT	STR	CTR	AVG
Baseline	90.89	78.06	74.89	81.28
$+\mathcal{L}_{\text{trip}}$	92.00	78.16	74.95	81.70
$+\mathcal{L}_{\text{char}}$	91.33	78.60	75.59	81.84
$+\mathcal{L}_{\text{adv}}$	**92.05**	**79.32**	**76.60**	**82.66**

"Baseline" denotes that the model is only trained by $\mathcal{L}_{\text{dec}}$, $\mathcal{L}_{\text{sty}}$, and $\mathcal{L}_{\text{cos}}$. The bold number denotes the best performance.

quality. Additionally, attempts to re-train this large-size model within the SM^{2}L framework (59.78 M vs. 4.96 M) led to non-convergence. Therefore, we cannot report the corresponding results here. These findings collectively underscore the superior adaptability and efficiency of our compact style transfer module.

4.5.3 Effect of the Multi-Task Learning. In the implementation of SM^{2}L, the optimization phase is driven by three distinct learning tasks, employing a variety of loss functions: triplet loss $\mathcal{L}_{\text{trip}}$, character loss $\mathcal{L}_{\text{char}}$, and adversarial loss $\mathcal{L}_{\text{adv}}$. Here, we conduct a series of experiments to demonstrate the effectiveness of this multi-task learning paradigm. It is noted that the baseline model for comparison predominantly utilizes $\mathcal{L}_{\text{dec}}$, $\mathcal{L}_{\text{sty}}$, and $\mathcal{L}_{\text{cos}}$ for optimization. The outcomes of these experiments, presented in Table 4, offer two significant insights: (1) Our baseline model, restricted to fundamental loss functions, attains a performance level comparable to a leading method cited in [38], as evidenced by similar average mAP scores across the three datasets (e.g., 81.28 for the Baseline vs. 79.36 for TD-STR). (2) The integration of a broader spectrum of loss functions notably boosts the performance of the SM^{2}L model. These results highlight the value of optimizing the retrieval network by considering both visual and semantic similarities.

4.5.4 Effect of the Detection Module. As mentioned in the main article, we employ the anchor-free general object detector, i.e., ATSS [49], as our detection module. This module, which tends to generate sufficient text proposals, benefits subsequent similarity ranking based on indefinite query texts. To verify the effectiveness of the detection module, we conduct a comparison experiment with a tailored ST detection module used in previous methods, i.e., **Fully Convolutional One-Stage Object Detector (FCOS)** [35], while other components remain unchanged. The quantitative results and illustrations are shown in Table 5 and Figure 8, which reveal that: (1) The ATSS-based SM^{2}L significantly outperforms the FCOS-based SM^{2}L across all metrics. (2) The FCOS-based SM^{2}L achieves better performance than the ATSS-based TD-STR (80.89 vs. 80.72), indicating the effectiveness of the proposed "T&I&I" paradigm. (3) The ATSS-based detection module provides more accurate proposals for small STs, such as "THE" in Figure 8(b), while the FCOS-based one can only generate

Table 5. Results of Different Detection Modules

Method	Detector	SVT	STR	CTR	AVG
TD-STR	FCOS	89.24	76.94	63.59	79.36
	ATSS	90.28	77.82	74.07	80.72
SM^2L	FCOS	90.43	77.28	74.96	80.89
	ATSS	**92.05**	**79.32**	**76.60**	**82.66**

The bold number denotes the best performance.

Fig. 8. The text image instance proposals generated by SM^2L with two detection modules.

several notable text proposals. Compelled by the above observations, our proposed detection module, profiting from the precise proposal generation, is well-suited for the ST retrieval task.

5 Conclusion

In this article, we propose SM^2L, a novel T&I&I approach encompassing three modality branches, specifically devised for ST retrieval tasks. Diverging from conventional T&I methods, our approach integrates a stylized middle modality and employs multi-task learning during the training phase to concurrently ensure visual and semantic coherence. This strategy results in notable performance enhancements without necessitating additional computational resources. Extensive experiments and ablation studies substantiate the superiority of SM^2L, consistently surpassing existing state-of-the-art methods in all metrics across three distinct datasets. We believe our study will serve as a strong baseline for future work and inspire more work in the line of SM2L for the ST retrieval task. Future work will focus on developing a more generalized method for multiple languages and arbitrarily oriented text.

References

[1] David Aldavert, Marçal Rusinol, Ricardo Toledo, and Josep Lladós. 2013. Integrating visual and textual cues for query-by-string word spotting. In *Proceedings of the IEEE International Conference on Document Analysis and Recognition*, 511–515.

[2] Jon Almazán, Albert Gordo, Alicia Fornés, and Ernest Valveny. 2014. Word spotting and recognition with embedded attributes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 12 (2014), 2552–2566.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*. Springer, 213–229.

[4] Xingping Dong and Jianbing Shen. 2018. Triplet loss in siamese network for object tracking. In *Proceedings of the European Conference on Computer Vision*, 459–474.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929. Retrieved from https://doi.org/10.48550/arXiv.2010.11929

[6] Shancheng Fang, Zhendong Mao, Hongtao Xie, Yuxin Wang, Chenggang Yan, and Yongdong Zhang. 2023. ABINet++: Autonomous, bidirectional and iterative language modeling for scene text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45 (2023), 7123–7141.

[7] Duoduo Feng, Xiangteng He, and Yuxin Peng. 2023. MKVSE: Multimodal knowledge enhanced visual-semantic embedding for image-text retrieval. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 5 (2023), 1–21.

[8] Zilong Fu, Hongtao Xie, Shancheng Fang, Yuxin Wang, Mengting Xing, and Yongdong Zhang. 2023. Learning pixel affinity pyramid for arbitrary-shaped text detection. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 1s (2023), 1–24.

[9] Lluís Gómez, Andrés Mafla, Marçal Rusinol, and Dimosthenis Karatzas. 2018. Single shot scene text retrieval. In *Proceedings of the European Conference on Computer Vision*, 700–715.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Communications of the ACM* 63, 11 (2020), 139–144.

[11] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the International Conference on Machine Learning*, 369–376.

[12] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. 2016. Synthetic data for text localisation in natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2315–2324.

[13] Xuan He, Zhiyong Li, Jiacheng Lin, Ke Nai, Jin Yuan, Yifan Li, and Runmin Wang. 2023. Domain adaptive multigranularity proposal network for text detection under extreme traffic scenes. *Computer Vision and Image Understanding* 233 (2023), 103709.

[14] Mingxin Huang, Jiaxin Zhang, Dezhi Peng, Hao Lu, Can Huang, Yuliang Liu, Xiang Bai, and Lianwen Jin. 2023. ESTextSpotter: Towards better scene text spotting with explicit synergy in transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19495–19505.

[15] Vladimir Iosifovich Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady* 10, 8 (1966), 707–710.

[16] Minghui Liao, Guan Pang, Jing Huang, Tal Hassner, and Xiang Bai. 2020. Mask textspotter v3: Segmentation proposal network for robust scene text spotting. In *Proceedings of the European Conference on Computer Vision*, 706–722.

[17] Yuliang Liu, Hao Chen, Chunhua Shen, Tong He, Lianwen Jin, and Liangwei Wang. 2020. Abcnet: Real-time scene text spotting with adaptive bezier-curve network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9809–9818.

[18] Yuliang Liu, Chunhua Shen, Lianwen Jin, Tong He, Peng Chen, Chongyu Liu, and Hao Chen. 2022. ABCNet v2: Adaptive Bezier-curve network for real-time end-to-end text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 11 (2022), 8048–8064.

[19] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. 2018. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In *Proceedings of the European Conference on Computer Vision*, 67–83.

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

[21] Anand Mishra, Karteek Alahari, and C. V. Jawahar. 2013. Image retrieval using textual cues. In *Proceedings of the IEEE International Conference on Computer Vision*, 3040–3047.

[22] Nibal Nayef, Yash Patel, Michal Busta, Pinaki Nath Chowdhury, Dimosthenis Karatzas, Wafa Khlif, Jiri Matas, Umapada Pal, Jean-Christophe Burie, Cheng-lin Liu, and Jean-Marc Ogier. 2019. ICDAR2019 robust reading challenge on multilingual scene text detection and recognition. In *Proceedings of the International Conference on Document Analysis and Recognition*, 1582–1587.

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of the Conference on Advanced Neural Information Processing Systems*, 8024–8035.

[24] Adrian Penate-Sanchez, David Freire-Obregon, Adrian Lorenzo-Melian, Javier Lorenzo-Navarro, and Modesto Castrillon-Santana. 2020. TGC20ReId: A dataset for sport event re-identification in the wild. *Pattern Recognition Letters* 138 (2020), 355–361.

[25] Yadong Qu, Qingfeng Tan, Hongtao Xie, Jianjun Xu, Yuxin Wang, and Yongdong Zhang. 2023. Exploring stroke-level modifications for scene text editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, 2119–2127.

[26] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.

[27] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 815–823.

[28] Ruth Schulz, Ben Talbot, Obadiah Lam, Feras Dayoub, Peter Corke, Ben Upcroft, and Gordon Wyeth. 2015. Robot navigation using human cues: A robot navigation system for symbolic goal-directed exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1100–1105.

[29] Jialie Shen, Marie Morrison, and Zhu Li. 2023. Scalable multimodal learning and multimedia recommendation. In *Proceedings of the IEEE International Conference on Collaboration and Internet Computing*. IEEE, 121–124.

[30] Jialie Shen, Meng Wang, Shuicheng Yan, and Xian-Sheng Hua. 2011. Multimedia tagging: past, present and future. In *Proceedings of the ACM International Conference on Multimedia*, 639–640.

[31] Baoguang Shi, Xiang Bai, and Cong Yao. 2016. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 11 (2016), 2298–2304.

[32] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556. Retreived from https://doi.org/10.48550/arXiv.1409.1556

[33] Sebastian Sudholt and Gernot A. Fink. 2016. Phocnet: A deep convolutional neural network for word spotting in handwritten documents. In *Proceedings of the International Conference on Frontiers in Handwriting Recognition*, 277–282.

[34] Shu Tian, Xu-Cheng Yin, Ya Su, and Hong-Wei Hao. 2017. A unified framework for tracking based text detection and recognition from web videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 3 (2017), 542–554.

[35] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9627–9636.

[36] Osman Tursun, Simon Denman, Rui Zeng, Sabesan Sivapalan, Sridha Sridharan, and Clinton Fookes. 2020. MTRNet++: One-stage mask-based scene text eraser. *Computer Vision and Image Understanding* 201 (2020), 103066.

[37] Andreas Veit, Tomas Matera, Lukas Neumann, Jiri Matas, and Serge Belongie. 2016. Coco-text: Dataset and benchmark for text detection and recognition in natural images. arXiv:1601.07140. Retreived from https://doi.org/10.48550/arXiv.1601.07140

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

[39] Kai Wang, Boris Babenko, and Serge Belongie. 2011. End-to-end scene text recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 1457–1464.

[40] Pengfei Wang, Chengquan Zhang, Fei Qi, Shanshan Liu, Xiaoqiang Zhang, Pengyuan Lyu, Junyu Han, Jingtuo Liu, Errui Ding, and Guangming Shi. 2021. PGNET: Real-time arbitrarily-shaped text spotting with point gathering network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2782–2790.

[41] Wenhai Wang, Enze Xie, Xiang Li, Xuebo Liu, Ding Liang, Zhibo Yang, Tong Lu, and Chunhua Shen. 2021. 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.

[42] Wenjia Wang, Enze Xie, Xuebo Liu, Wenhai Wang, Ding Liang, Chunhua Shen, and Xiang Bai. 2020. Scene text image super-resolution in the wild. In *Proceedings of the European Conference on Computer Vision*, 650–666.

[43] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

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

[45] Gengshen Wu, Zijia Lin, Jungong Han, Li Liu, Guiguang Ding, Baochang Zhang, and Jialie Shen. 2018. Unsupervised deep hashing via binary latent factor models for large-scale cross-modal retrieval. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Vol. 1, 5.

[46] Hongtao Xie, Shancheng Fang, Zheng-Jun Zha, Yating Yang, Yan Li, and Yongdong Zhang. 2019. Convolutional attention networks for scene text recognition. *ACM Transactions on Multimedia Computing, Communications, and Applications* 15, 1s (2019), 1–17.

[47] Liang Xie, Jialie Shen, Jungong Han, Lei Zhu, and Ling Shao. 2017. Dynamic multi-view hashing for online image retrieval. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Vol. 78, 122.

[48] Liang Xie, Jialie Shen, and Lei Zhu. 2016. Online cross-modal hashing for web image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30, 294–300.

[49] Zecheng Xie, Yaoxiong Huang, Yuanzhi Zhu, Lianwen Jin, Yuliang Liu, and Lele Xie. 2019. Aggregation cross-entropy for sequence recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6538–6547.

[50] Song Yang, Qiang Li, Wenhui Li, Xuan-Ya Li, Ran Jin, Bo Lv, Rui Wang, and Anan Liu. 2023. Semantic completion and filtration for image–text retrieval. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 4 (2023), 1–20.

[51] Maoyuan Ye, Jing Zhang, Shanshan Zhao, Juhua Liu, Tongliang Liu, Bo Du, and Dacheng Tao. 2023. Deepsolo: Let transformer decoder with explicit points solo for text spotting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19348–19357.

[52] Zheng-Jun Zha, Meng Wang, Jialie Shen, and Tat-Seng Chua. 2012. Text mining in multimedia. *Mining Text Data* (2012), 361–384.

[53] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z. Li. 2020. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9759–9768.

[54] Wei Zhang, Ting Yao, Shiai Zhu, and Abdulmotaleb El Saddik. 2019. Deep learning–based multimedia analytics: A review. *ACM Transactions on Multimedia Computing, Communications, and Applications* 15, 1s (2019), 1–26.

[55] Xiang Zhang, Yongwen Su, Subarna Tripathi, and Zhuowen Tu. 2022. Text spotting transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9519–9528.

[56] Zuoyan Zhao, Hui Xue, Pengfei Fang, and Shipeng Zhu. 2024. PEAN: A diffusion-based prior-enhanced attention network for scene text image super-resolution. In *Proceedings of ACM International Conference on Multimedia*.

[57] Suping Zhou, Jia Jia, Yufeng Yin, Xiang Li, Yang Yao, Ying Zhang, Zeyang Ye, Kehua Lei, Yan Huang, and Jialie Shen. 2019. Understanding the teaching styles by an attention based multi-task cross-media dimensional modeling. In *Proceedings of the ACM International Conference on Multimedia*, 1322–1330.

[58] Lei Zhu, Jialie Shen, Liang Xie, and Zhiyong Cheng. 2016. Unsupervised visual hashing with semantic assistant for content-based image retrieval. *IEEE Transactions on Knowledge and Data Engineering* 29, 2 (2016), 472–486.

[59] Shipeng Zhu, Zuoyan Zhao, Pengfei Fang, and Hui Xue. 2023. Improving scene text image super-resolution via dual prior modulation network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3843–3851.

Received 11 December 2023; revised 13 July 2024; accepted 3 September 2024