CLIPTER: Looking at the Bigger Picture in Scene Text Recognition

Aviad Aberdam[1*] David Bensaïd[2†] Alona Golts[1] Roy Ganz[2†] Oren Nuriel[1]
Royee Tichauer[1] Shai Mazor[1] Ron Litman[1]

[1]AWS AI Labs [2]Technion, Israel

Abstract

Reading text in real-world scenarios often requires understanding the context surrounding it, especially when dealing with poor-quality text. However, current scene text recognizers are unaware of the bigger picture as they operate on cropped text images. In this study, we harness the representative capabilities of modern vision-language models, such as CLIP, to provide scene-level information to the crop-based recognizer. We achieve this by fusing a rich representation of the entire image, obtained from the vision-language model, with the recognizer word-level features via a gated cross-attention mechanism. This component gradually shifts to the context-enhanced representation, allowing for stable fine-tuning of a pretrained recognizer. We demonstrate the effectiveness of our model-agnostic framework, CLIPTER (CLIP TExt Recognition), on leading text recognition architectures and achieve state-of-the-art results across multiple benchmarks. Furthermore, our analysis highlights improved robustness to out-of-vocabulary words and enhanced generalization in low-data regimes.

1. Introduction

Recognizing text in real-world settings often involves leveraging contextual information from the scene, particularly when dealing with blurry, low-resolution, corrupted, or occluded text, as showcased in Fig. 1. Conversely, learning-based methods typically detect text in the image and then perform recognition solely on the cropped detected regions, neglecting valuable scene information [8, 38, 20, 9, 2, 1, 6, 48, 11, 39, 70, 72, 46]. As a result, the practice of operating on cropped text images is inherently suboptimal.

To overcome this limitation, we explore the use of vision-language models. These models, pretrained on a vast corpus of image-caption pairs, exhibit powerful representation capabilities and can be used for numerous downstream tasks [51, 63, 44, 14, 4, 34, 65, 35]. Unlike models trained only on visual data, such as MAE [23], vision-language models are also supervised by the corresponding textual description. This description brings focus to the crucial details in the scene, which in turn can assist in reading poor-quality text, as we show later. Moreover, the image caption can even contain actual text words in the image, such as busi-

*Corresponding author aaberdam@amazon.com.
†Work done during an Amazon internship.

Figure 1: **The Importance of Seeing the Bigger Picture.** Scene context often assists in reading text in real-world scenarios, and in certain cases, it is even vital. Thus, current crop-based text recognizers are inherently limited (Top). To address this limitation, our method, CLIPTER, provides the recognizer with scene information (Bottom).

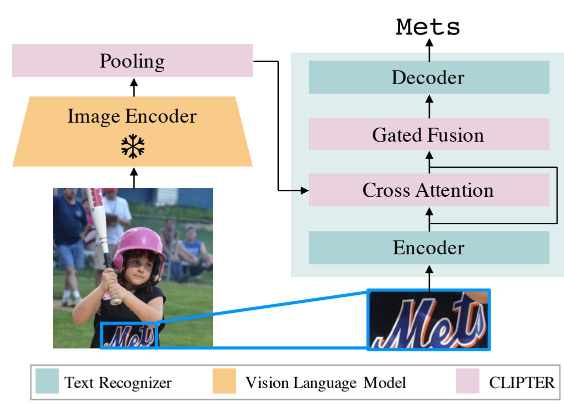

Figure 2: **CLIPTER – Incorporating Scene Context into Text Recognizers.** Our novel approach employs a frozen vision-language model, such as CLIP, to extract rich features of the entire scene image. These features are then fused with the crop-level features using our gated cross-attention mechanism, which gradually shifts the pretrained recognizer to the context-enriched features.

ness logos and street names, due to their necessity for describing the scene. Hence, leveraging vision-language models can facilitate in recognizing such words, which are typically unique, categorized as out-of-vocabulary, and therefore pose greater difficulty to text recognition models [60].

In this work, we introduce CLIPTER (CLIP TExt Recognition), a general framework for integrating image-level knowledge into crop-based text recognizers. To this end, our method first extracts a rich visual representation of the entire image using a vision-language image encoder. As depicted in Fig. 2, this representation is then merged with the word-level features of the cropped text instance using a cross-attention-based operation. Additionally, we incorporate a gating mechanism, which gradually shifts between the word-level features and the merged representations during training. This mechanism provides a more stable training process and enables the adaptation of pre-existing models, including those pretrained on synthetic data. As a result, CLIPTER can effectively enhance any pretrained recognizer with scene context awareness.

We design our method as a versatile framework consisting of modular blocks of varying sizes that can support various text recognition architectures and adapt to diverse computational constraints. In particular, we explore a range of vision and vision-language image encoders, pooling operators, light-to-heavy fusion schemes, and different integration points between word-level and image-level representations. This integration point is critical and highly dependent on the underlying architecture, and therefore we study two types of integration point: early fusion within the vision model, which considers the image representation as additional visual content, and late fusion at the decoding stage, which utilizes the image features as supplementary contextual information to condition the prediction on.

Throughout extensive experimentation on twelve highly-diverse datasets, our method exhibits consistent improvements on top of various leading text recognition methods, such as TRBA [8], ABINet [20], and PARSeq [11]. In particular, implementing CLIPTER on PARSeq achieves state-of-the-art (SoTA) results on all benchmarks, including dense text and challenging street-view images. Further in-depth analysis reveals that incorporating CLIPTER improves robustness to out-of-vocabulary words and enhances generalization capability in low-data regimes.

To account for all the computations involved in adding CLIPTER to an existing recognizer, we perform an end-to-end evaluation, in which we cascade the recognizer after an existing text detector. This setting not only reveals performance gains over two-stage and end-to-end approaches, but also demonstrate a marginal impact in the overall latency. Finally, through a comprehensive ablation study, we develop a recipe for integrating CLIPTER in other text recognition architectures, including future ones.

To summarize, our main contributions are:

- Introducing CLIPTER, a framework for enhancing text recognition performance by incorporating scene context through the use of vision-language models.
- The design of a computationally efficient and flexible framework that can be incorporated with various existing text recognition architectures.
- Demonstrating consistent improvements of leading text recognizers on diverse datasets, achieving state-of-the-art results, and enhancing robustness to out-of-vocabulary and generalization in low-data regimes.

2. Related Work

Scene text recognition. Significant progress has been made in word-level scene text recognition in recent years [8, 38, 50, 9, 48, 64, 67, 70, 56, 13], largely due to the adoption of transformer-based models [6, 11, 20, 46] and the exploitation of unlabeled data [3, 39, 43, 1, 72, 42]. Recent SoTA method includes ViTSTR [7] and MaskOCR [43], which propose simple ViT-based architectures to improve vision extraction, and ABINet [20] and SRN [67], which incorporate linguistic knowledge through transformer-based language modalities to refine vision model predictions. Additionally, SeqCLR [3], CCR [72], Persec [39] and SemiMTR [1] use contrastive learning and consistency regularization to learn from unlabeled data. Nevertheless, all these methods suffer from a lack of scene-level context, as they operate on cropped text images. Consequently, in challenging cases of corrupted text, these models resort to predict the most likely word from their training vocabulary [60, 21]. We address this limitation by enriching the recognizer with scene-level information.

It should be noted that while there is an alternative end-to-end approach called text spotting [37, 33, 73, 73, 66, 26], which allows the text decoder to access the entire image when decoding the text, our work focuses on improving the cascaded pipeline of separate detection and recognition steps. The cascaded pipeline, which is widely used and studied, offers advantages such as modularity, task decoupling, invariance to scale and rotation [53], and efficiency in using synthetic data [8].

Vision-language models. Vision-language models trained on a large number of image-text pairs provide effective representations for various tasks [68, 51, 35, 65, 34, 25, 61, 16, 63]. Among the pioneers in this area, CLIP [51] used contrastive learning to train image and text encoders to produce aligned representations of image-caption pairs. BLIP [34] proposed a filtering mechanism to handle noisy image-caption pairs, while GIT [61] simplified the architecture to only one image encoder and one text decoder. Inspired by their potential, we aim to utilize them for scene text recognition.

3. Methodology

Our method proposes a model-agnostic and easy-to-implement framework, which is designed to compensate for the lack of scene context in crop-based text recognizers. In this section, we detail the building blocks, describe the training procedure, discuss the running time, and present several recognition models on which we apply our method.

3.1. Building Blocks

Our algorithm consists of four elements. First, the image encoder generating the image-level features. Next, an optional pooling operation on the obtained features can reduce memory and latency. In the third stage, an integration point is determined to incorporate these features into the target text recognizer. Finally, a fusion mechanism merges the image- and word-level representations. The algorithm pseudocode is provided in Appendix A and its implementation on PARSeq architecture [11] is illustrated in Fig. 3.

Image Encoder The image encoder aims to complement the recognizer word-level information with scene context-aware representations. We explore several powerful encoders, which can be divided into two categories:

- *Vision-based models* such as ViT [19], MAE [23] and DiNO [15], which are pretrained solely on images and encompass the image visual content, including the class and position information of its objects.
- *Vision-language-based models* such as CLIP [51], BLIP [34] and GiT [61], which are pretrained on a massive and highly-diversified dataset of images and their textual descriptions. These descriptions focus the representations on the crucial details in the scene, and as shown in Sec. 6, lead to better performance.

Our study focuses on transformer-based [19] vision encoders, where the number of output representations corresponds to the number of image patches, denoted as HW, plus an additional representation for the special token `[class]`. To maintain its generalization ability and prevent a substantial increase in training runtime and memory, we keep the image encoder frozen during training.

Image Feature Pooling The size of the image features affects training and inference times, as they are integrated with word-level representations using a cross-attention-based mechanism. The computational complexity of this operation is $\mathcal{O}(N_{\text{global}} \cdot N_{\text{local}} \cdot d)$, where N_{global} and N_{local} denote the corresponding sequence lengths of the image-level and word-level representations, and d is their dimension.

To optimize the balance between performance and latency, a pooling component is introduced. This layer preserves the first image-level representation corresponding to the `[class]` token and applies 2D average pooling to the remaining per-patch representations. As a result, the output feature size becomes $\mathbf{F}^{\text{global}} \in \mathbb{R}^{(1+HW/k^2)\times d}$, where k denotes the pooling kernel. In Sec. 6, we empirically study the trade-off between computational cost and granularity level in the choice of k. Surprisingly, our findings reveal that even using only the first representation of CLIP (marked as $k=\infty$) can still improve performance.

Integration Point Another decision is where to inject the global, image-level features within the recognition model. Since recognition architectures differ significantly, we presume there is no fixed integration point and thus explore several options for each recognizer. In Sec. 3.3, we describe the studied recognition schemes and define optional integration points for each. However, in general, these integration points can be divided into two main categories:

- *Early fusion* – The integration is performed in the vision encoder and targets the visual features extracted by a convolutional, or transformer-based backbone. This approach views the global features as visual content and therefore merges them with the visual features of the crop. In some architectures, there are several options for an early integration point.
- *Late fusion* – The integration is performed in the linear, attention-based or transformer-based decoder. This fusion can be seen as conditional decoding, in which the characters are decoded given the image state. In autoregressive decoders, such integration leads to a significant increase in inference time, repeating the cross-attention operation in each decoding step.

Note that early and late fusion approaches have been studied in the literature [36, 28, 5, 35], although not in our context of merging local and global information in text recognition.

Fusion Mechanism The role of this component is fusing the image-level and word-level features, $\mathbf{F}^{\text{global}}$ and $\mathbf{F}^{\text{local}}$, correspondingly. To this end, we first linearly project the global features to the dimension of the local representations, d. Then, we choose between two attention-based schemes:

- *Multi-head cross-attention (MH-CA)* – The nowadays natural approach of combining two data streams of different resources [58, 18]. In our case, queries are local features, and global features are the keys and values:

$$\mathbf{F}^{\text{mixed}} = \text{MH-CA}(Q{=}\mathbf{F}^{\text{local}}, K{=}\mathbf{F}^{\text{global}}, V{=}\mathbf{F}^{\text{global}}).$$

  We examine several, compact to heavy, models with a different number of attention heads, hidden layers, hidden sizes, and intermediate sizes.

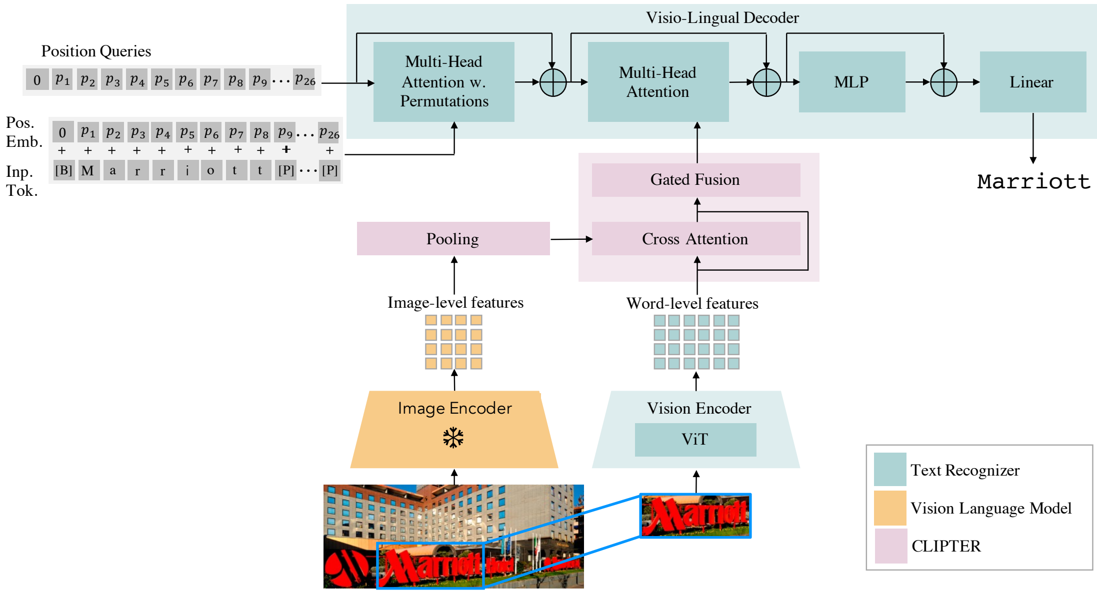

Figure 3: **An Overview of CLIPTER Integrated into PARSeq [11].** Our framework introduces four building blocks: (1) A pretrained *Image Encoder* used to extract high-level representations from the entire image. (2) These representations are then fed to the *Image Feature Pooling* layer that can pool spatial dimensions, balancing latency and performance. (3) Upon obtaining features, an *Integration Point* (early or late) is chosen to incorporate this information, which can change for different architectures. (4) Lastly, the *Fusion Mechanism* composed of a gated cross-attention mechanism, merges the two streams of information allowing the recognizer to reason over both.

- *Gated attention* – A lightweight alternative that applies a gated mechanism between global and local features. Such a model cannot handle different lengths of representation sequences. Therefore, it can be utilized only if there is a single image-level representation after the pooling ($k = \infty$). In this case, for each local representation $\mathbf{f}_i^{\text{local}} \in \mathbb{R}^d$ we independently apply:

$$\mathbf{g} = \text{softmax}(\mathbf{W}\left[\mathbf{f}_i^{\text{local}}; \mathbf{f}^{\text{global}}\right]) \quad (1)$$

$$\mathbf{f}_i^{\text{mixed}} = \mathbf{g} \circ \mathbf{f}_i^{\text{local}} + (\mathbf{1} - \mathbf{g}) \circ \mathbf{f}^{\text{global}}, \quad (2)$$

where $\mathbf{W} \in \mathbb{R}^{d \times 2d}$ is a weight matrix, and $\circ$ is an elementwise Hadamard product.

Our training starts with a pretrained baseline model, which we fine-tune to become context-aware. To enhance the stability of this process, we implement a tanh-gating mechanism inspired by [24, 4]. This mechanism preserves the forward-pass intact during initialization and gradually transitions between the original word-level features and the fused representation throughout training:

$$\mathbf{F}^{\text{fused}} = (1 - \tanh(\alpha))\mathbf{F}^{\text{local}} + \tanh(\alpha)\mathbf{F}^{\text{mixed}}, \quad (3)$$

where α is a learnable scalar initialized at 0.

3.2. Training Protocol

The CLIPTER framework is a versatile solution that can be used for various text recognition schemes. Instead of adjusting the training parameters for each recognition platform, we employ a pretrained baseline model to initialize all the original parts. Then, we only fine-tune the text recognizer and the fusion mechanism, with hyperparameters that are agnostic to the chosen recognition architecture. This approach reduces the training time and allows for the utilization of synthetic data in training the baseline model.

As the image encoder is fixed in our approach, we can save training time and memory by preparing the dataset in advance. This involves pre-computing the image-level representations by passing all training images through the image encoder. The resulting dataset can be used for training different recognition architectures. To further reduce data loading latency, we cache the image-level representations during the first epoch of training. These measures lead to a minor increase in the training time of each iteration, less than 10% when using a reasonable fusion mechanism.

3.3. Studied Recognition Models

Here, we detail at a high-level the text recognition models we examine, as well as suggested integration points.

- *TRBA* – A general architecture [8], which comprises four components: (i) transformation for normalization of the input image, (ii) a ResNet-based visual feature extractor, (iii) a Bi-LSTM-based contextual block, and (iv) an attention decoder. We explore three integration points: *visual* and *contextual*, corresponding to early

Method	SVT 647	IC13 757	IC15 2,077	COCO 5,716	RCTW 962	Uber 49,561	ArT 3,677	LSVT 3,911	RECTS 2,219	MLT19 4,100	TextOCR 70,597	HierText 75,829	Average 220,053	Weighted Average
TRBA [8]	94.9	98.5	84.8	79.2	81.1	80.5	89.2	77.9	90.4	90.7	82.9	85.1	86.3	83.4
+ CLIPTER $_{\text{Vision}}$	95.4	98.8	85.3	79.3	81.3	82.0	90.2	79.4	91.1	91.1	83.9	85.8	87.0	84.3
Δ	**+0.5**	**+0.3**	**+0.5**	**+0.1**	**+0.2**	**+1.5**	**+1.0**	**+1.5**	**+0.7**	**+0.4**	**+1.0**	**+0.7**	**+0.7**	**+0.9**
ViTSTR-S [6]	92.3	97.0	81.8	77.0	72.9	77.4	86.9	73.7	88.5	89.4	80.4	83.2	83.4	80.9
+ CLIPTER $_{\text{Vision}}$	93.4	97.1	82.3	77.7	75.3	79.6	88.2	76.0	89.5	89.9	81.8	84.0	84.6	82.3
Δ	**+1.1**	**+0.1**	**+0.5**	**+0.7**	**+2.4**	**+2.2**	**+1.3**	**+2.3**	**+1.0**	**+0.5**	**+1.4**	**+0.8**	**+1.2**	**+1.4**
ABINet-Vis [20]	88.4	97.0	80.7	75.8	72.2	75.8	85.5	72.0	87.4	89.0	78.7	83.0	82.1	79.8
+ CLIPTER $_{\text{Vision}}$	91.8	97.0	82.0	77.1	76.1	78.1	87.4	74.9	87.8	89.4	80.6	84.4	83.9	81.5
Δ	**+3.4**	0.0	**+1.3**	**+1.3**	**+3.9**	**+2.3**	**+1.9**	**+2.9**	**+0.4**	**+0.4**	**+1.9**	**+1.4**	**+1.8**	**+1.7**
ABINet [20]	**96.6**	97.6	85.1	79.4	76.7	80.8	89.2	76.6	89.4	90.2	83.1	86.6	85.9	83.9
+ CLIPTER $_{\text{Decoder}}$	96.0	98.3	85.4	79.3	78.6	82.1	89.3	77.1	89.7	90.2	83.4	86.7	86.3	84.3
Δ	-0.6	**+0.7**	**+0.3**	-0.1	**+1.9**	**+1.3**	**+0.1**	**+0.5**	**+0.3**	0	**+0.3**	**+0.1**	**+0.4**	**+0.4**
PARSeq [11]	96.1	98.9	85.7	80.5	81.4	83.2	91.2	80.2	**91.8**	91.5	85.2	87.4	87.8	85.6
+ CLIPTER $_{\text{Vision}}$	**96.6**	**99.1**	**85.9**	**81.0**	**82.1**	**84.4**	**91.7**	**81.8**	**91.8**	**91.6**	**86.0**	**88.0**	**88.3**	**86.4**
Δ	**+0.5**	**+0.2**	**+0.2**	**+0.5**	**+0.7**	**+1.2**	**+0.5**	**+1.6**	0	**+0.1**	**+0.8**	**+0.6**	**+0.5**	**+0.8**

Table 1: **Results on the Scene Text Benchmarks.** Scene text recognition accuracy (%) over twelve public benchmarks. The number of words in each dataset is listed below its name, and we report average and weighted average results. Integrating CLIPTER into top-performing recognition architectures consistently improves performance. In particular, CLIPTER advances the state-of-the-art performance of PARSeq [11] by +0.5% and +0.8% on average and weighted average, respectively.

fusion after stages (ii) and (iii), and *decoder*, of fusing representations in the prediction block of stage (iv).

- *ViT-STR* – This scheme consists of a single stage, and hence, the integration point is at the output of the ViT.
- *ABINet* – A multimodal scheme [20], comprising three components: (i) a vision model with a ResNet backbone network and transformer unit, (ii) a language model that refines the output vision embeddings, and (iii) a fusion model that combines the output of the vision and language models for final prediction. We investigate two early integration points within the vision model (stage i): *visual* after the ResNet backbone, and *contextual* after the transformer unit, as well as a late fusion: *decoder* within the fusion model (stage iii).
- *PARSeq* – A transformer-based architecture [11], depicted in Fig. 3. Here we define an early integration: *visual* after the ViT model, and late integration: *decoder* after the cross attention block.

4. Experiments

We hereby present a comprehensive evaluation of our approach in combination with state-of-the-art text recognizers on twelve diverse scene text recognition benchmarks. Our study showcases the broad applicability of our proposed method, as it consistently outperforms existing approaches across all datasets and architectures. Moreover, we conduct an in-depth analysis of our method's generalization capability on images containing out-of-vocabulary words, revealing a significant improvement in performance. Finally, we demonstrate that our approach surpasses current methods in scenarios with limited amounts of labeled training data.

Datasets. We follow the pre-processing of [10, 1] and similarly perform experiments using only real data. We demonstrate our results on twelve scene text benchmarks: ICDAR 2013 [31], ICDAR 2015 [29], ArT [17], SVT [62], LSVT [57], COCO-Text [59], RCTW [54], Uber [71], ReCTS [69], MLT19 [47], HierText [41] and TextOCR [55]. Both TextOCR and HierText are particularly large datasets, rich with text and containing 30 and 100 words on average per image, respectively. In total, our test set contains over 200k words, 20 times larger than similar works [38, 8, 20, 3, 48]. Dataset characteristics and train/test splits are provided in Appendix B. We evaluate performance using word-level accuracy and normal/weighted averages across datasets.

Implementations Details. We adopt the codebase of PARSeq[1] [11] and SemiMTR[2] [1] to establish our baselines. Our experiments are conducted on 4 Tesla V100 GPUs, 16GB memory, using PyTorch. We closely follow the configuration parameters of the baseline models, as detailed in Appendix C. We use a 36-character set (10 digits and 26 letters) in most experiments, except for using the full 94-character set in Sec. 4.2. We introduce a range of cross-attention models, including *gated attention* and three multi-head cross-attention (MH-CA) types: *tiny*, *mini* and *small*, containing 328K, 923K, 5.3M and 18.1M parameters, respectively, and are further elaborated in Appendix C.

4.1. Improving State-of-the-Art Recognizers

In Tab. 1, we examine the impact of CLIPTER on leading scene text recognition methods across 12 diverse benchmarks. Despite the diversity in architectures, including CNN-based and transformer-based visual encoders, and autoregressive and parallel decoders, we demonstrate

[1] https://github.com/baudm/parseq
[2] https://github.com/amazon-science/semimtr-text-recognition

Figure 4: **Qualitative Examples.** The eight images on the left depict failure cases of the baseline model PARSeq [11], that become success cases when incorporating CLIPTER . On the right, we present examples where our method produces incorrect predictions and where both models fail to correctly decode the text.

that CLIPTER significantly improves performance for all tested methods. In particular, the improvements in accuracy weighted average are $+0.9\%$ in TRBA, $+1.4\%$ in ViT-STR, $+1.7\%$ in ABINet-VIS, and $+0.4\%$ in ABINet. Moreover, we establish new SoTA results by integrating CLIPTER with PARSeq, the current top-performing text recognizer, increasing its accuracy by $+0.8\%$ across all datasets. In Fig. 4, we present qualitative results of successful and unsuccessful cases, with additional examples in Appendix G.

Breaking down the evaluation datasets reveals that our method is especially beneficial for street-view datasets, namely Uber, SVT and LSVT, decreasing the relative error of PARSeq by almost 10%. Uber-Text data [71] is predominantly comprised of street names and business logos, presenting multiple challenging text instances that are blurry, occluded, or of low-resolution. In such cases, the surrounding context plays a vital role, even for human perception. Furthermore, our method exhibits improvements on text-rich datasets, TextOCR and HierText, with an average of 30 and 100 words per image [41], compared to less than 10 words in other datasets. These results demonstrate that leveraging image-level information can be advantageous even for text-dense images and documents, indicating the potential of vision-language models to reason from text in images. See further analysis in Appendix F.3.

As expected, we observe that a one-size-fits-all approach is not feasible due to the vast differences in the architectures of text recognizers. Therefore, for each recognition scheme, we present the best results achieved using CLIPTER and leave the discussion of design choices to Sec. 6. More precisely, in Tab. 1, we denote the integration point in subscript, use the MH-CA mini for the fusion mechanism, and the image encoder is BLIP (with pooling of $k = 5$) for ABINet and CLIP ($k = \infty$) for the others.

Method	OOV 25,647	IV 91,191	Average 116,838
PARSeq [11]	68.97	79.74	77.38
+ CLIPTER $_{\text{Vision}}$	**71.45**	**80.99**	**78.9**
Δ	**+2.48**	**+1.25**	**+1.52**

Table 2: **Out-Of-Vocabulary.** Our method not only leads to an improvement over in-vocabulary words, but also to a significant boost on out-of-vocabulary words.

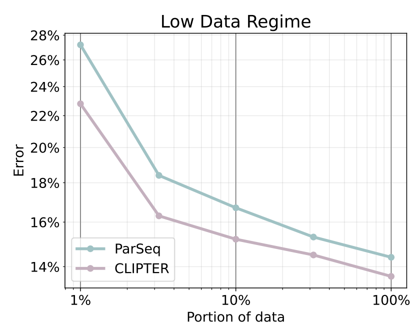

Figure 5: **Word Error Rate Versus Data Portion in Log-Log Scale.** Our method, trained on 40% of the data, reaches the baseline performance when trained on the entire data.

4.2. Performance on Out-Of-Vocabulary

Motivated by the improved results on street-view images, we examine our method on out-of-vocabulary (OOV) text instances – words that do not appear in the training sets. These are often crucial for understanding the scene, as they can contain prices, names, dates, phone numbers, emails, and URLs. However, as shown in [60, 21], current methods over-rely on their train vocabulary, especially in low-

	Method	ICDAR 2015		Total-Text	
		E2E (G)	FPS	Word-Spotting (None)	FPS
E2E	ABCNet v2 [40]	73.0	6.5	70.4	-
	Mask TextSpotter v3 [37]	74.2	2.6	-	2.2
	MANGO [49]	73.9	4.3	72.9	4.3
	GLASS [53]	76.3	**7.75**	79.9	**7.5**
2-STG	GLASS + PARSeq [11]	77.3	6.7	79.8	6.3
	GLASS + CLIPTER	**77.4**	6.2	**80.6**	5.9

Table 3: **E2E Text Spotting.** We compare end-to-end (E2E) methods against two-stage (2-STG) pipelines that use GLASS for text detection and PARSeq, with and without CLIPTER, for text recognition. Although CLIPTER increases E2E latency by 10 ms per image, it improves SoTA results of GLASS by **+0.9** on IC15 and **+0.7** on Total-Text.

Method	**Average** 220,053	**Weighted Average**
TRBA [8]	86.28	83.38
+ $\text{CLIPTER}_{\text{Lightweight}}$	**+0.43**	**+0.71**
ViTSTR-S [6]	83.37	80.89
+ $\text{CLIPTER}_{\text{Lightweight}}$	**+1.16**	**+1.28**
PARSeq [11]	87.76	85.65
+ $\text{CLIPTER}_{\text{Lightweight}}$	**+0.58**	**+0.69**

Table 4: **CLIPTER Lightweight.** Even in its lightweight version, consisting only of a $\text{CLIP}_{\text{base}}$ image encoder and gated attention, CLIPTER enhances text recognizers.

quality or distorted text. To test if scene context can assist in these cases, we utilize the newly proposed OOV benchmark [21]. As demonstrated in Tab. 2, integrating CLIPTER into PARSeq, not only improves accuracy by 1.52% on general words, but is even more significant in OOV words, presenting an improvement of 2.48%. The robustness to OOV words is yet another reason to harness the knowledge of massively pretrained vision-language models.

4.3. Performance in Low Data Regime

To further probe the benefits of our method, we evaluate its performance in the low data regimes of 1%, 5%, 10%, and 25% of the training data. As shown in Fig. 5, CLIPTER leverages the generalization power of the vision-language model and thus becomes even more effective and beneficial when the amount of training data decreases. In particular, training CLIPTER on 10% of the data leads to similar results as the baseline trained on 25%. Likewise, when training CLIPTER on 40%, it achieves the performance of the baseline trained on 100% of the data. Similar trends appear with TRBA and ViTSTR, as shown in Appendix D.

5. End-to-End Latency and Performance

To accurately measure the impact of our method on latency, we aim to account for all its components and recognize that the encoding of the entire image is computed only once, regardless of the number of words it contains. Therefore, instead of calculating latency of standalone recognition on a single cropped text image, we construct an end-to-end evaluation that better simulates real-world latency. For this purpose, we employ the text detector of GLASS [53] and cascade it with PARSeq, both with and without CLIPTER. Here, we focus on a lightweight version of our method that consists of $\text{CLIP}_{\text{base}}$ image encoder and gated attention fusion mechanism. Our results, as shown in Tab. 3, indicate that our method adds only 8% to the overall latency (+12 ms per image) while delivering superior performance that outperforms both two-stage pipelines and existing E2E text spotting methods. For completeness, we present the recognition results of the lightweight version in Tab. 4, demonstrating nearly optimal performance, and offer further implementation details in Appendix E.

		Image Encoder	k	**Feature Shape**	**Average** 220,053	**Weighted Average**
PARSeq	**Bl.**	–	–	–	87.76	85.65
	Vis.	DINO [15]	2	50×768	+0.46	+0.58
		ViT-MAE [23]	–	$50 \times 1,024$	+0.44	+0.59
		OWL-ViT [44]	4	37×768	+0.55	+0.67
	Vis.-Lan.	GIT_{L} [61]	4	$17 \times 1,024$	+0.53	+0.75
		BLIP [34]	5	37×768	+0.56	+0.74
		$\text{CLIP}_{\text{base}}$ [51]	∞	1×512	+0.69	+0.71
		CLIP [51]	∞	1×768	**+0.73**	**+0.82**
TRBA	**Bl.**	–	–	–	86.28	83.38
	Pooling	BLIP [34]	∞	1×768	+0.15	+0.12
		BLIP [34]	10	10×768	**+0.60**	**+0.87**
		BLIP [34]	5	37×768	+0.55	+0.80
		BLIP [34]	3	101×768	**+0.60**	+0.84

Table 5: **Image Encoder and Pooling.** Word accuracy for different pretrained image encoders, as well as pooling kernel sizes. **Bl.** stands for baseline

6. Ablation Studies

Here, we study the relative effect of each component in our scheme, including choice of image encoder, pooling kernel size, integration point and fusion mechanism. Throughout our analysis, we discuss the performance-latency tradeoff and provide general recommendations for integrating CLIPTER in other text recognition methods.

The Choice of the Image Encoder. The first part of Tab. 5 exhibits the performance of PARSeq with CLIPTER, when leveraging the vision-based image encoders of DiNO, ViT-MAE and OWL-ViT, and when using the vision-language models of CLIP, BLIP and GIT. As shown, the best performance is achieved with the latter models. These models were pretrained not only on images, but also on their textual descriptions, leading to more informative and effective representations. Interestingly, the compact representation of CLIP leads to the best results in PARSeq. This, however, is not the case for all recognizers. For example, ABINet benefits more from the larger representation of BLIP. In general,

Method	**Average** 220,053	**Weighted Average**
TRBA [8]	86.28	83.38
+ CLIPTER$_{\text{Vision}}$	+0.67	**+0.95**
+ CLIPTER$_{\text{Contextual}}$	+0.59	+0.9
+ CLIPTER$_{\text{Decoder}}$	**+0.72**	+0.9
ViTSTR-S [6]	83.37	80.89
+ CLIPTER$_{\text{Vision}}$	**+1.2**	**+1.36**
ABINet-Vis [20]	82.14	79.75
+ CLIPTER$_{\text{Vision}}$	**+1.73**	**+1.76**
+ CLIPTER$_{\text{Contextual}}$	+1.14	+0.97
ABINet [20]	85.85	83.85
+ CLIPTER$_{\text{Vision}}$	+0.3	+0.18
+ CLIPTER$_{\text{Contextual}}$	+0.18	+0.36
+ CLIPTER$_{\text{Decoder}}$	**+0.49**	**+0.5**
PARSeq [11]	87.76	85.65
+ CLIPTER$_{\text{Vision}}$	+0.55	**+0.76**
+ CLIPTER$_{\text{Decoder}}$	**+0.56**	+0.71

Table 6: **Integration Point.** In each text recognizer, there are several integration points to fuse the image and crop-level features. The results indicate that the optimal point depends on the recognizer architecture.

though, the go-to method is still the single representation of CLIP, as it yields nearly the best performance and demonstrates low computation cost.

In the second part of Tab. 5, we examine the effect of the pooling kernel k. To this end, we apply CLIPTER on TRBA with a fixed image encoder, BLIP, and only change the kernel size. As shown, too aggressive pooling ($k = \infty$) deteriorates representation quality. However, besides this extreme, varying k does not impact performance significantly but can lead to severe consequences on running times. Thus, our recommendation is to use a relatively coarse representation of the scene, which performs decently well.

Integration Point. In Tab. 6, we evaluate the effect of the integration point on our studied architectures, considering the three types defined in Sec. 3: *vision* and *contextual* in early fusion, and *decoder* in late fusion. As shown, TRBA and PARSeq are less sensitive to this decision; whereas ABINet benefits from late fusion and ABINet-Vis from early, vision fusion. Note, however, that the runtime complexity of late fusion increases dramatically for autoregressive decoders, as in PARSeq and TRBA, but not for parallel decoders, as in ABINet. As expected, the significant differences between the text recognition architectures imply that the decision of the integration point is not clear-cut and thus, integrating CLIPTER in new architectures require an empirical search to locate the optimal fuse point.

Fusion Mechanism. We examine the effect of the fusion model capacity, considering a compact gated attention scheme to more complex multi-head attention modules.

	Fusion Mechanism	**Image Encoder**	**Recog. GFLOPS**	**Average** 220,053	**Weighted Average**
TRBA	–	–	6.681	86.28	83.38
	Gated attention	CLIP	+0.005	+0.62	+0.82
	MH-CA tiny	CLIP	+0.019	+0.50	+0.82
	MH-CA tiny	BLIP$_{k=5}$	+0.033	+0.54	+0.77
	MH-CA mini	CLIP	+0.126	**+0.67**	**+0.95**
	MH-CA mini	BLIP$_{k=5}$	+0.185	+0.56	+0.86
	MH-CA small	CLIP	+0.502	+0.54	+0.81
	MH-CA small	BLIP$_{k=5}$	+0.620	+0.52	+0.81
ViTSTR	–	–	4.608	83.37	80.89
	Gated attention	CLIP	+0.002	+1.22	+1.35
	MH-CA tiny	CLIP	+0.004	**+1.25**	+1.34
	MH-CA tiny	BLIP$_{k=5}$	+0.016	+1.16	+1.29
	MH-CA mini	CLIP	+0.024	+1.2	**+1.36**
	MH-CA mini	BLIP$_{k=5}$	+0.086	+1.21	+1.29
	MH-CA small	CLIP	+0.109	+1.16	+1.28
	MH-CA small	BLIP$_{k=5}$	+0.223	+1.11	+1.26
PARSeq	–	–	3.174	87.76	85.65
	Gated attention	CLIP	+0.039	+0.53	+0.71
	MH-CA tiny	CLIP	+0.081	+0.54	+0.71
	MH-CA tiny	BLIP$_{k=5}$	+0.098	+0.5	+0.7
	MH-CA mini	CLIP	+0.533	+0.55	+0.76
	MH-CA mini	BLIP$_{k=5}$	+0.599	+0.56	+0.74
	MH-CA small	CLIP	+2.005	**+0.63**	**+0.77**
	MH-CA small	BLIP$_{k=5}$	+2.137	+0.54	+0.72

Table 7: **Effect of the Fusion Mechanism.** Word accuracy when using Gated Attention or Multi-Headed Cross-Attention (MH-CA) fusion mechanisms. Note that the GFLOPS count refers to the recognition operation only.

Since gated-attention can be applied only for a single vector, we examine it with CLIP image encoder ($k = \infty$). For the other alternatives, we also consider the spatial representations of BLIP with $k = 5$. As shown in Tab. 7, more complex schemes usually lead to better results, but at the cost of additional computational load, represented by the number of FLOPS. We find the gated-attention and MH-CA mini as balancing points between quality and runtime.

7. Conclusions

We introduced CLIPTER, a novel approach to enrich crop-based text recognizers with scene knowledge, by utilizing vision-language models. Our versatile framework is composed of modular blocks, which enable the fine-tuning of various pretrained text recognition architectures. Our extensive experiments on diverse benchmarks demonstrate that incorporating CLIPTER into existing approaches consistently enhances their performances, demonstrating better generalization and robustness to out-of-vocabulary words. Moreover, our end-to-end evaluation revealed a marginal increase in the overall latency, while presenting improved results, even compared to text spotting methods. Finally, through a comprehensive ablation study, we provide guidelines for implementing our method on future recognizers, paving the way for further advancements in this area.

References

[1] Aviad Aberdam, Roy Ganz, Shai Mazor, and Ron Litman. Multimodal semi-supervised learning for text recognition. *arXiv preprint arXiv:2205.03873*, 2022. 1, 2, 5, 17

[2] Aviad Aberdam, Ron Litman, Shahar Tsiper, Oron Anschel, Ron Slossberg, Shai Mazor, R Manmatha, and Pietro Perona. Sequence-to-sequence contrastive learning for text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15302–15312, 2021. 1

[3] Aviad Aberdam, Ron Litman, Shahar Tsiper, Oron Anschel, Ron Slossberg, Shai Mazor, R Manmatha, and Pietro Perona. Sequence-to-sequence contrastive learning for text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15302–15312, 2021. 2, 5

[4] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 1, 4

[5] Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R Manmatha. Docformer: End-to-end transformer for document understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 993–1003, 2021. 3

[6] Rowel Atienza. Vision transformer for fast and efficient scene text recognition. In *International Conference on Document Analysis and Recognition*, pages 319–334. Springer, 2021. 1, 2, 5, 8, 16

[7] Rowel Atienza. Vision transformer for fast and efficient scene text recognition. In *International Conference on Document Analysis and Recognition*, pages 319–334. Springer, 2021. 2

[8] Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. What is wrong with scene text recognition model comparisons? dataset and model analysis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4715–4723, 2019. 1, 2, 4, 5, 8, 16

[9] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. What if we only use real datasets for scene text recognition? toward scene text recognition with fewer labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3113–3122, 2021. 1, 2

[10] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. What if we only use real datasets for scene text recognition? toward scene text recognition with fewer labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3113–3122, 2021. 5, 13

[11] Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. In *Proceedings of the 17th European Conference on Computer Vision (ECCV)*, Cham, 10 2022. Springer International Publishing. 1, 2, 3, 4, 5, 6, 7, 8, 18

[12] Prajjwal Bhargava, Aleksandr Drozd, and Anna Rogers. Generalization in nli: Ways (not) to go beyond simple heuristics, 2021. 16

[13] Ayan Kumar Bhunia, Pinaki Nath Chowdhury, Aneeshan Sain, and Yi-Zhe Song. Towards the unseen: Iterative text recognition by distilling from errors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14950–14959, 2021. 2

[14] Ali Furkan Biten, Ron Litman, Yusheng Xie, Srikar Appalaraju, and R Manmatha. Latr: Layout-aware transformer for scene-text vqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16548–16558, 2022. 1

[15] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021. 3, 7

[16] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. 2

[17] Chee Kheng Chng, Yuliang Liu, Yipeng Sun, Chun Chet Ng, Canjie Luo, Zihan Ni, ChuanMing Fang, Shuaitao Zhang, Junyu Han, Errui Ding, et al. Icdar2019 robust reading challenge on arbitrary-shaped text-rrc-art. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1571–1576. IEEE, 2019. 5, 13

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3, 16

[19] 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. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 3

[20] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: autonomous, bidirectional and iterative language modeling for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7098–7107, 2021. 1, 2, 5, 8

[21] Sergi Garcia-Bordils, Andrés Mafla, Ali Furkan Biten, Oren Nuriel, Aviad Aberdam, Shai Mazor, Ron Litman, and Dimosthenis Karatzas. Out-of-vocabulary challenge report. *arXiv preprint arXiv:2209.06717*, 2022. 2, 6, 7, 16

[22] 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. 17, 18

[23] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable

vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1, 3, 7

[24] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 4

[25] Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. Scaling up vision-language pre-training for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17980–17989, 2022. 2

[26] Mingxin Huang, Yuliang Liu, Zhenghao Peng, Chongyu Liu, Dahua Lin, Shenggao Zhu, Nicholas Yuan, Kai Ding, and Lianwen Jin. Swintextspotter: Scene text spotting via better synergy between text detection and text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4593–4603, 2022. 2

[27] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *arXiv preprint arXiv:1406.2227*, 2014. 17, 18

[28] Hamid Reza Vaezi Joze, Amirreza Shaban, Michael L Iuzzolino, and Kazuhito Koishida. Mmtm: Multimodal transfer module for cnn fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13289–13299, 2020. 3

[29] 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. 5

[30] 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. 13

[31] 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. 5

[32] 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. 13

[33] Yair Kittenplon, Inbal Lavi, Sharon Fogel, Yarin Bar, R Manmatha, and Pietro Perona. Towards weakly-supervised text spotting using a multi-task transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4604–4613, 2022. 2

[34] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022. 1, 2, 3, 7

[35] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 1, 2, 3

[36] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 3

[37] Minghui Liao, Guan Pang, Jing Huang, Tal Hassner, and Xiang Bai. Mask textspotter v3: Segmentation proposal network for robust scene text spotting. In *European Conference on Computer Vision*, pages 706–722. Springer, 2020. 2, 7

[38] Ron Litman, Oron Anschel, Shahar Tsiper, Roee Litman, Shai Mazor, and R Manmatha. Scatter: selective context attentional scene text recognizer. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11962–11972, 2020. 1, 2, 5

[39] Hao Liu, Bin Wang, Zhimin Bao, Mobai Xue, Sheng Kang, Deqiang Jiang, Yinsong Liu, and Bo Ren. Perceiving stroke-semantic context: Hierarchical contrastive learning for robust scene text recognition. AAAI, 2022. 1, 2

[40] 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. *arXiv preprint arXiv:2105.03620*, 2021. 7

[41] Shangbang Long, Siyang Qin, Dmitry Panteleev, Alessandro Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1049–1059, 2022. 5, 6, 13

[42] Canjie Luo, Lianwen Jin, and Jingdong Chen. Siman: Exploring self-supervised representation learning of scene text via similarity-aware normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1039–1048, 2022. 2

[43] Pengyuan Lyu, Chengquan Zhang, Shanshan Liu, Meina Qiao, Yangliu Xu, Liang Wu, Kun Yao, Junyu Han, Errui Ding, and Jingdong Wang. Maskocr: Text recognition with masked encoder-decoder pretraining. *arXiv preprint arXiv:2206.00311*, 2022. 2

[44] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection with vision transformers. *arXiv preprint arXiv:2205.06230*, 2022. 1, 7

[45] Anand Mishra, Karteek Alahari, and CV Jawahar. Scene text recognition using higher order language priors. 2012. 13

[46] Byeonghu Na, Yoonsik Kim, and Sungrae Park. Multimodal text recognition networks: Interactive enhancements between visual and semantic features. In *European Confer-*

ence on Computer Vision, pages 446–463. Springer, 2022. 1, 2

[47] 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. 5, 13

[48] Oren Nuriel, Sharon Fogel, and Ron Litman. Textadain: Fine-grained adain for robust text recognition. *arXiv preprint arXiv:2105.03906*, 2021. 1, 2, 5

[49] Liang Qiao, Ying Chen, Zhanzhan Cheng, Yunlu Xu, Yi Niu, Shiliang Pu, and Fei Wu. MANGO: A mask attention guided one-stage scene text spotter. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, issue 3, pages 2467–2476, 2021. 7

[50] Zhi Qiao, Yu Zhou, Dongbao Yang, Yucan Zhou, and Weiping Wang. Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13528–13537, 2020. 2

[51] 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. 1, 2, 3, 7

[52] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048, 2014. 13

[53] Roi Ronen, Shahar Tsiper, Oron Anschel, Inbal Lavi, Amir Markovitz, and R Manmatha. Glass: Global to local attention for scene-text spotting. *arXiv preprint arXiv:2208.03364*, 2022. 2, 7, 17

[54] Baoguang Shi, Cong Yao, Minghui Liao, Mingkun Yang, Pei Xu, Linyan Cui, Serge Belongie, Shijian Lu, and Xiang Bai. Icdar2017 competition on reading chinese text in the wild (rctw-17). In *2017 14th iapr international conference on document analysis and recognition (ICDAR)*, volume 1, pages 1429–1434. IEEE, 2017. 5, 16

[55] Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8802–8812, 2021. 5, 16

[56] Ron Slossberg, Oron Anschel, Amir Markovitz, Ron Litman, Aviad Aberdam, Shahar Tsiper, Shai Mazor, Jon Wu, and R Manmatha. On calibration of scene-text recognition models. *arXiv preprint arXiv:2012.12643*, 2020. 2

[57] Yipeng Sun, Jiaming Liu, Wei Liu, Junyu Han, Errui Ding, and Jingtuo Liu. Chinese street view text: Large-scale chinese text reading with partially supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9086–9095, 2019. 5, 13

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[59] 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. 5, 13

[60] Zhaoyi Wan, Jielei Zhang, Liang Zhang, Jiebo Luo, and Cong Yao. On vocabulary reliance in scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11425–11434, 2020. 2, 6

[61] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022. 2, 3, 7

[62] 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. 5, 16

[63] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv preprint arXiv:2202.03052*, 2022. 1, 2

[64] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. From two to one: A new scene text recognizer with visual language modeling network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14194–14203, 2021. 2

[65] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021. 1, 2

[66] 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. 2

[67] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12113–12122, 2020. 2

[68] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Making visual representations matter in vision-language models. 2021. 2

[69] Rui Zhang, Yongsheng Zhou, Qianyi Jiang, Qi Song, Nan Li, Kai Zhou, Lei Wang, Dong Wang, Minghui Liao, Mingkun Yang, et al. Icdar 2019 robust reading challenge on reading chinese text on signboard. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 1577–1581. IEEE, 2019. 5, 16

[70] Xinyun Zhang, Binwu Zhu, Xufeng Yao, Qi Sun, Ruiyu Li, and Bei Yu. Context-based contrastive learning for scene text recognition. AAAI, 2022. 1, 2

[71] Ying Zhang, Lionel Gueguen, Ilya Zharkov, Peter Zhang, Keith Seifert, and Ben Kadlec. Uber-text: A large-scale

dataset for optical character recognition from street-level imagery. In *SUNw: Scene Understanding Workshop-CVPR*, volume 2017, page 5, 2017. 5, 6, 16, 17

[72] Caiyuan Zheng, Hui Li, Seon-Min Rhee, Seungju Han, Jae-Joon Han, and Peng Wang. Pushing the performance limit of scene text recognizer without human annotation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14116–14125, 2022. 1, 2

[73] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2