GLASS: Global to Local Attention for Scene-Text Spotting

Roi Ronen[1]*, Shahar Tsiper[2]*, Oron Anschel[2],
Inbal Lavi[2], Amir Markovitz[2], and R. Manmatha[2]

[1] Viterbi Faculty of Electrical & Computer Eng., Technion, Haifa, Israel
[2] AWS AI Labs

Abstract. In recent years, the dominant paradigm for text spotting is to combine the tasks of text detection and recognition into a single *end-to-end* framework. Under this paradigm, both tasks are accomplished by operating over a shared global feature map extracted from the input image. Among the main challenges that end-to-end approaches face is the performance degradation when recognizing text across scale variations (smaller or larger text), and arbitrary word rotation angles. In this work, we address these challenges by proposing a novel global-to-local attention mechanism for text spotting, termed GLASS, that fuses together global and local features. The global features are extracted from the shared backbone, preserving contextual information from the entire image, while the local features are computed individually on resized, high resolution rotated word crops. The information extracted from the local crops alleviates much of the inherent difficulties with scale and word rotation. We show a performance analysis across scales and angles, highlighting improvement over scale and angle extremities. In addition, we introduce an orientation-aware loss term supervising the detection task, and show its contribution to both detection and recognition performance across all angles. Finally, we show that GLASS is general by incorporating it into other leading text spotting architectures, improving their text spotting performance. Our method achieves state-of-the-art results on multiple benchmarks, including the newly released TextOCR.

Keywords: Text spotting, Text Detection, Text Recognition, Language Understanding.

1 Introduction

Text spotting, the task of detecting text instances in the wild and recognizing them, has seen a notable increase in performance in recent years. It is now commonly used in many real-life scenarios and applications. Demanding areas such as autonomous driving, document analysis, and geo-localization, where accurate text transcription is a must, all rely on text spotting. The challenge lies in the

* Equal contribution. For correspondence: `tsiper@amazon.com`
Code available at https://www.github.com/amazon-research/glass-text-spotting

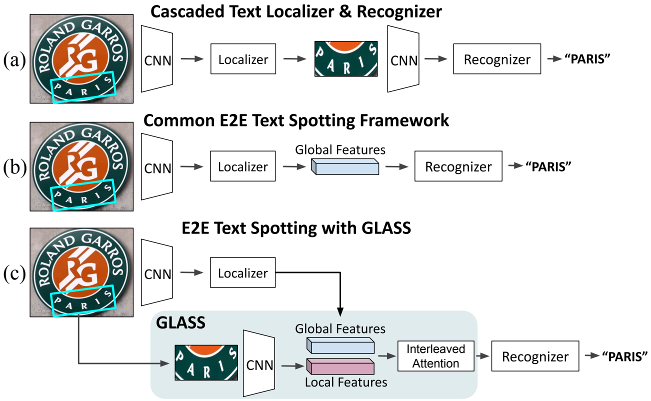

Fig. 1. An overview of text spotting approaches. **(a)** Cascaded. A standalone text detector followed by a standalone recognizer. Each is trained separately. **(b)** End-to-end (E2E) text spotting. Detection and recognition are jointly optimized. **(c)** Our approach with GLASS fusion, operating on two separate feature maps taken at different resolutions and contexts, bridging (a) and (b). Feature maps are fused using interleaved attention, improving robustness to scale and rotation, and overall performance.

fact that some words may span the entire image, while other words, even in the same image, may be hard to read, e.g., appear on a traffic sign barely seen from across the street.

Two prevalent paradigms exist for text spotting (see Fig. 1): the first is a modular approach, cascading independent text detection and recognition models. The recognition model uses uniformly aligned and resized word-crop images as input with upright orientation, abstracting away scale and rotation. The components in this approach are mostly explored independently in the literature, isolating either the word detection performance (ignoring transcripts) [47,4,2,21,39], or the recognition performance over datasets composed of word-crop images [1,41,25,31]. The second approach is a combined End-to-End (E2E) architecture, adding a recognition branch that operates directly on the detection model's latent features [8,3,16,36,33,20,29]. Feature sampling replaces cropping, allowing detection and recognition to be jointly trained E2E.

With E2E becoming the common paradigm, scale and rotation-free crops were often replaced by sampled CNN features, that are highly sensitive to both scale and rotation [13,46]. While the joint optimization in E2E systems improved performance for average-sized and upright-facing words, scale extremities and strong rotations were overlooked.

In this work, we propose to bridge the two paradigms to get the best out of both worlds. We combine *global* features from the detector's embedding space with *local* cropped word embeddings. The fusion is done using a novel Global-to-Local interleaved attention module, leveraging information from both feature maps. This global-to-local approach enriches the information used by the recognition branch, boosting the overall text spotting accuracy for different scales and rotations. Additional gains for rotated text are obtained by introducing an orientation prediction side-task, aimed at better capturing rotated words, or words in rotated images. The side task is supervised by a new loss term with a πn periodic sine-squared function. The model is optimized end-to-end, benefiting both detection and recognition. We name our approach GLASS - Global to Local Attention for Scene-text Spotting.

Our method achieves state-of-the-art results on ICDAR 2015 [14], Total-Text [6], and Rotated ICDAR 2013 [20] benchmarks. We also present blind evaluation results measured on the recently released TextOCR [42] dataset, largely surpassing the baseline. GLASS is then examined across a range of text scales and orientations in an ablation study. Finally, we incorporate GLASS into recent E2E text spotting approaches, and show gains of 2.3% for Mask TextSpotter v3 [20] and 3.7% for ABCNet v2 [29], when measuring E2E F-score on Total-Text [6]. To summarize, the main contributions of this work are:

1. A new global-to-local attention module improving text spotting performance at scale extremities
2. A periodic orientation loss, further improving detection and recognition results across all angles
3. State-of-the-art results on ICDAR 2015 [14], Total-Text [6], TextOCR [42] and Rotated ICDAR 2013 [20] benchmarks
4. Incorporation of GLASS into other text spotting frameworks, demonstrating consistent gains

2 Background and Related Work

Text Spotting We compare the two paradigms for text spotting, cascaded and E2E. The cascaded option enjoys modularity, allowing to combine different architectures for detection and recognition. By uniformly scaling and rotating the word crops to their upright orientation, the recognizer is operating on a fixed and less challenging input space. Another benefit is that each part can train using different data. The recognizer can leverage large amounts of synthetically generated word-crops, tailored for specific lexicons and challenging scenarios [12,19], which cannot be leveraged by the detector. For detection, synthetic images are largely limited to pre-training [10,30]. The main caveat in the cascaded approach is that no contextual information is shared between the predicted words during recognition.

In contrast, in E2E methods the recognizer leverages contextual information from each word's surroundings, which helps disambiguate and overcome challenging scenarios. This is due to the large receptive field of CNN backbones [32]. Furthermore, jointly training detection and recognition, benefits both tasks [38,45], leading to substantial gains. Finally, such methods often enjoy improved latency, since the feature extraction step is done once, and shared by the detector and the recognizer. A main drawback of E2E approaches is the limited resolution at which the recognizer operates. The recognition branch is commonly fed with sampled features at a fixed spatial size [20,29,3,36], which might be insufficient for accurate prediction [31,41,25]. Specifically, the feature sampling operator, which provides the input features to the recognizer, is lossy and may fail to preserve meaningful information. The sampling procedure is sensitive to different text scales and orientations, as discussed in ABCNet v2 [29] and shown below.

Feature Sampling As recognition operates on features sampled from a latent space, the sampling procedure plays a large role in its success. Different sampling approaches have seen several advancements over the years.

Region of Interest (RoI) Pooling [9] was first introduced for sampling features, and has been widely used since [18,45]. It was replaced with RoIAlign [11] that used a bilinear interpolation for weighted feature sampling, that was also extended for the first time for sampling non axis-aligned (i.e., *rotated*) RoIs [27]. For sampling arbitrarily shaped text, further extensions [33,16,20] added a background mask to the sampling operation for isolating the extracted word only, often relying on segmentation-based detectors or masks.

For text, Mask TextSpotter v3 [20] presented an anchor-free, non-parametric, segmentation proposal network where original detections are in the form of a segmentation map. Features were sampled using hard RoI masking. Recently, Liu *et al.* introduced ABCNet [28] and ABCNet v2 [29] which use a Bézier curve parametrization for localizing curved text. They exploited the parametrization using a BezierAlign operator for feature sampling.

In the above methods, the text recognition module operates only on the limited resolution features pooled from the whole-image, the global feature map. Our method is the first to combine additional information computed directly from a normalized word crop. Since it is not tailored to a specific backbone or pooling layer, GLASS can be added on top of multiple existing E2E frameworks, as we show in Sec. 4.4.

A few notable works predict text without relying on feature sampling. These include CharNet [26], which directly outputs bounding boxes of words and characters with corresponding character labels, and MANGO [36], which divides the input image into grids and coarsely localizes the text sequence using a position-aware attention module.

Global-to-Local Fusion There have been approaches in the literature for improving object detection performance across a large range of scales. Recent approaches [22,43,44] focused on fusing between different layers of the shared feature extraction backbone. They harness the fact that different layers at different depths within the shared backbone have a different receptive field, and are capable of detecting details at a multitude of scales. We leverage the global-to-local key concept from object detection [22,43] and adopt it for the recognition task in E2E text spotting.

Orientation Prediction Several recent works modeled the text detection problem using a rotated box geometry for the detections. Among the first was EAST [47], that suggested a hybrid approach for regressing both a rotated box

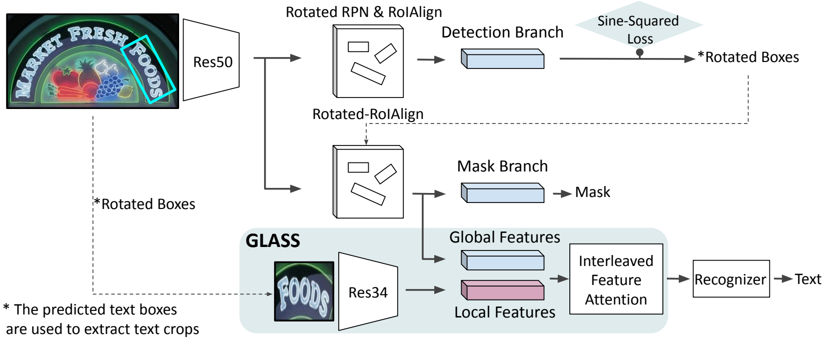

Fig. 2. Global to Local Scene-text Spotting. The global detection branch is a Mask R-CNN variant supervised by our novel sine-squared loss for rotated box prediction. During inference, predicted boxes are used to sample global backbone features and to crop the original image as input for the local feature extractor. Global and local features are fused in GLASS and passed to the recognizer for transcript extraction.

and a quadrilateral around text objects. The use of rotated RPN proposals and Rotated-ROIAlign was first suggested in [34], using the regular L_1 loss to regress the output boxes. The authors in [48], identified an ambiguity in the angle prediction, namely that the same box can be described by four valid angles, a different angle perpendicular for each face. They tackled it with a cascaded process where a single correct box orientation is regressed in a gradual manner. In [17] the same ambiguity was handled by optimizing over the minimal angle difference among all of the detected box sides, and in [35] the ambiguity is dealt with by representing the orientation of each box with 8 parameters, and regressing over all of them, while ensuring continuity of the loss function. Our approach tackles the angle regression ambiguity by introducing an orientation-aware, periodic trigonometric loss, as further discussed in Sec. 3.2.

3 Method

3.1 GLASS Fusion Module

Our pipeline is composed of three principal components, seen in Fig. 2: the detection branch, the GLASS fusion module, and the recognition head. The detection branch is used for locating words, predicting their bounding boxes and segmentation masks. It is trained with the added orientation-aware loss, and its backbone is used for extracting the global features. The fusion module combines the global and local features, producing an enriched embedding that is then fed into the recognition head. In this work, we use a Rotated Mask R-CNN [11,34] as the baseline approach for our detection branch, and for the recognition branch we use ASTER [41].

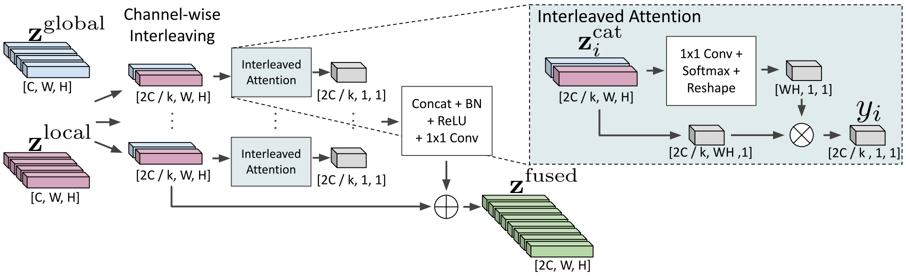

Fig. 3. Global to Local attention feature fusion. GLASS takes both *global* (image) and *local* (crop-level) features as input, and outputs the fused feature map. The input feature maps are channel-wise interleaved, concatenated and split into k blocks. Each block is processed by an attention module, producing k tensor outputs. These are then concatenated, transformed by 1×1 convolution and summed element-wise with the input feature maps.

We begin by presenting our global-to-local fusion module in Sec. 3.1, and follow with a description of our orientation-aware loss in Sec. 3.2. Finally, we discuss aspects regarding the overall architecture and training objective in Sec. 3.3.

We propose a fusion module for incorporating the scale and rotation invariance of the local word-crop approach into an end-to-end text spotter, while still using global context. Uniformly scaled and aligned crops abstract away nuisances such as the original word's size and off-axis rotation, and allow us to maximize our ability to extract text.

The predicted boxes of the detection branch are used in two sampling operations. For *global* features, we sample the FPN [23] features from the detection branch. For *local* features, we sample the image directly (i.e. crop), performing an affine transformation that yields a uniformly scaled, axis-aligned word crop. This crop is passed through a local feature extractor.

Formally, we denote the input image $\mathbf{x}$ and its FPN features $\mathbf{z}$. Following detection, the global feature map $\mathbf{z}$ is sampled using the predicted boxes, yielding $\mathbf{z}^{\text{global}}$. Using the same boxes, the image $\mathbf{x}$ is cropped and aligned into $\mathbf{x}^{\text{local}}$, which in turn is embedded using a shallow dedicated backbone into $\mathbf{z}^{\text{local}}$. Every text detection is now represented by two separate feature maps, $\mathbf{z}^{\text{global}}$ and $\mathbf{z}^{\text{local}}$, illustrated in Fig. 3 by light-blue and pink bars, respectively.

Inspired by [31], we propose an interleaved attention procedure that operates over small feature blocks, aiming to combine and use the most relevant information for the text recognition task, from both input features. Attending over small blocks is significantly lighter than standard attention mechanisms in high dimensionality, and is shown to improve robustness for downstream tasks [31]. The interleaved attention combines global and local features in a learned way, maximizing informational content. This dynamic weighting allows the attention mechanism to place greater emphasis on specific relevant context, depending on the input.

The global and local features are first combined to k block tensors by an interleaved concatenation, where $k \ll C$. The ith block is given by

$$\mathbf{z}_i^{\text{cat}} = \left[z_{i \cdot m+1}^{\text{global}},\ z_{i \cdot m+1}^{\text{local}},\ \ldots,\ z_{i \cdot m+m}^{\text{global}},\ z_{i \cdot m+m}^{\text{local}} \right], \tag{1}$$

where $m = \lceil C/k \rceil$. Block indices are given by $i \in \{0, 1, \ldots, k-1\}$, and z_j^{global}, z_j^{local} depict the jth channel of $\mathbf{z}^{\text{global}}$, $\mathbf{z}^{\text{local}}$, for $j \in [1, C]$.

A spatial attention operator is then applied to each of the k blocks in $\mathbf{z}^{\text{cat}}$, as shown in Fig. 3 within the dashed box, such that

$$y_i = \text{vec}(\mathbf{z}_i^{\text{cat}})^T \text{vec}\left(\text{softmax}(v_i^T \mathbf{z}_i^{\text{cat}})\right), \tag{2}$$

yielding an attentional vector $y_i \in \mathbb{R}^{2C/k}$. Here $v_i \in \mathbb{R}^{2C/k}$ is a learnable vector and vec$(\cdot)$ reshapes a tensor of size (C, W, H) into a matrix of size (C, WH). Interleaving the two feature maps ensures that in Eq. (2) we mix information that corresponds with both global and local features.

Next we stack the k attention vectors $y_{1 \ldots k}$ channel-wise, apply batch normalization (BN), ReLU and a 1 × 1 convolution for capturing channel-wise dependencies, resulting in the tensor $\mathbf{y}$. The fused output is an element-wise addition of $\mathbf{y}$ and the interleaved-concatenated feature maps, $\mathbf{z}^{\text{cat}}$, illustrated by the green bars in Fig. 3. Formally,

$$\mathbf{z}^{\text{fused}} = \mathbf{z}^{\text{cat}} + \mathbf{w}^T \, \text{ReLU}(\text{BN}(\mathbf{y})), \tag{3}$$

where $\mathbf{w} \in \mathbb{R}^{C \times H \times W}$ are learnable weights. The output $\mathbf{z}^{\text{fused}}$ is then used as input to the recognition head.

We note that there are two alternatives to the proposed interleaved attention. The first is a naïve concatenation of the global and local features, which is tested in Sec. 3.2, and is shown to reduce accuracy. The second is performing a full attention computation between the full dimension of local and global features, however, this computation is unfeasible due to computational limitations.

3.2 Orientation Prediction

Unlike objects in other common object-detection benchmarks such as COCO [24] and Pascal VOC [7], text instances are long, narrow and directed. A word extracted upside-down or rotated by 90° is usually non-recoverable in terms of recognition. This makes orientation prediction especially important and meaningful. To this end, we propose a new orientation-aware loss function operating on rotated box detections $\mathbf{r} \in \mathbb{R}^{N \times 5}$, where the first 4 coordinates describe the rotated box center, width and height, and the last coordinate, $\theta \in \mathbb{R}$, depicts its upward facing angle. The loss function for the mth matched detection is given by

$$\mathcal{L}_{\text{rbox}} = \sum_{i=1}^{4} \alpha_i \left| \hat{r}_i - r_i \right| + \alpha_\theta \sin^2\left(\hat{\theta} - \theta\right), \tag{4}$$

where the hat denotes prediction. The constants α_i for $i \in [1,4]$ and α_θ are chosen empirically. The sine-squared function has a periodicity of $n\pi$ for $n \in \mathbb{Z}$, leveraging the fact that a rotated rectangle is symmetric to $n\pi$ rotations. This symmetry removes an inherent ambiguity during training, allowing the same prediction for boxes that are either upright or flipped upside-down. This mechanism was empirically shown to improve detection results across all angular range, as further explored in Sec. 4.5 and shown in Fig. 4. For each word, the orientation angle is then used to perform a rotated pooling operation on the shared backbone features, yielding the global feature input. This process is common in additional E2E frameworks [29,20], but without the orientation-aware loss, orientation mistakes in the form of discrete jumps of $k\pi/2, k \in \mathbb{Z}$ degrees are more common. In our implementation, the predicted angle is also used to generate an oriented word-crop, from which the local features are computed, as illustrated in Fig. 2.

3.3 Global to Local End-to-end Text Spotting

Here, we describe our proposed E2E framework with GLASS, shown in Fig. 2. For the shared backbone, we use the commonly used ResNet50 and FPN [23]. Its associated features $\mathbf{z}^{\text{global}}$ are sampled using Rotated-RoIAlign operating directly on the FPN levels as in [38]. For obtaining the local feature maps $\mathbf{z}^{\text{local}}$, we first sample a crop of the text RoI from the input image using Rotated-RoIAlign layer. Then, the crop features are extracted by ResNet34 backbone [5].

Finally, we fuse the global and local feature maps using the interleaved attention operation, described in Sec. 3.1, yielding $\mathbf{z}^{\text{fused}}$, which is the recognition module's input. The recognition head, detailed in the Supplementary Material, provides the transcript for each word. We note that the mask head is used as a parallel branch to the recognizer, and only receives $\mathbf{z}^{\text{global}}$ as input.

The overall loss function $\mathcal{L}$ used for the E2E supervised training is given by

$$\mathcal{L} = \mathcal{L}_{\text{rbox}} + \lambda_1 \mathcal{L}_{\text{mask}} + \lambda_2 \mathcal{L}_{\text{rec}} \; . \tag{5}$$

Here, the mask loss $\mathcal{L}_{\text{mask}}$ is identical to Mask R-CNN [11] and $\mathcal{L}_{\text{rec}}$ is the recognition loss.

We note that GLASS has no effect on the computational cost of the detector, including its mask branch, and the recognizer head. The computational aspects of GLASS, as well as the loss terms used in training, are further discussed in the Supplementary Material.

4 Experiments

We evaluate the performance of our method on several benchmarks, testing our method's robustness to rotations and text size. First, we compare our full framework with GLASS to current art. Next, we examine GLASS when integrated into two common E2E text spotting architectures, Mask TextSpotter v3 [20] and

ABCNet v2 [29]. Finally, we provide a comprehensive ablation study isolating the contribution of GLASS for different data distributions with various settings. Additional ablation studies are presented in the Supplementary Material.

4.1 Datasets

SynthText [10] is a synthetically generated dataset containing approximately 800K images and 6M synthetic text instances. **ICDAR 2013** [15] has 233 testing images containing mostly horizontal text. We synthetically rotate these images by various angles and use it to measure our performance on rotated text. **ICDAR 2015** [14] consists of 1,000 training images and 500 testing images. Most of the images are of low resolution and contain small text instances. **Total-Text** [6] contains 1,255 training and 300 testing images. It offers text instances in a variety of shapes, including horizontal, rotated, and curved text. **TextOCR** [42] is a recently published arbitrary-shaped detection and recognition dataset containing of 21,778 train, 3153 validation and 3232 test images with more than 700k, 100k and 80k annotated words, respectively.

4.2 Implementation details

We follow the common SynthText pre-training scheme [20,3]. For Total-Text, we fine-tune using a mixture of Total-Text and SynthText datasets, as in [3]. For ICDAR13 and ICDAR15 results, we train also on both datasets, following [20]. For TextOCR results, we follow the baseline [42] and use all of the datasets mentioned in Sec. 4.1. In the ablation studies (Sec. 4.5), we fine-tune the model for 100k iterations with a batch size of 8 images. In Sec. 4.3, the model is fine-tuned for 250k iterations with a batch size of 24 images. The recognizer used is an off-the-shelf component, based on ASTER [41]. Additional implementation details are found in the Supplementary Material.

4.3 Comparison with State-of-the-Art

Quantitative results for end-to-end text recognition on the ICDAR15, Total-Text and TextOCR datasets are listed in Table 1. For ICDAR15, our method outperforms previously reported word spotting protocol results for all three lexicons, and for the end-to-end evaluation protocol with Generic lexicon. For the Total-Text dataset, our method achieves state-of-the-art F-measure results for both settings in the word spotting evaluation, and for full-lexicon end-to-end. For no-lexicon end-to-end, GLASS outperforms all methods but CRAFTS [3].

We are among the first to report results for the challenging TextOCR test dataset in Table 1. This newly released dataset is an order of magnitude larger than previous ones and has ample variation in scale and rotation. Thresholds for the detection and recognition heads were set using the TextOCR validation set. Our method achieves state-of-the-art F-measure results, surpassing Mask TextSpotter v3 [20] by 16.3% on end-to-end evaluation protocol. Both methods were optimized with a similar data profile, including TextOCR train data [42].

Table 1. Results for ICDAR 2015, Total-Text and TextOCR datasets. 'S', 'W' and 'G' refer to strong, weak and generic lexicons. "None" refers to recognition without any lexicon. "Full" lexicon contains all the words in the test set. (*) refers to using specific lexicons from [20]. (†) indicates IoU of 0.1 was used instead of 0.5 during evaluation. (‡) represents results obtained using method's official source code.

Method	ICDAR 2015						Total-Text				TextOCR
	Word Spotting			End-to-End			Word Spotting		End-to-End		End-to-End
	S	W	G	S	W	G	None	Full	None	Full	
TextDragon [8]	86.2	81.6	68.0	82.5	78.3	65.2	-	-	48.8	71.8	-
ABCNet v2 [29]	-	-	-	82.7	78.5	73.0	70.4	78.1	-	-	-
MTSv3* [20]	83.1	79.1	75.1	83.3	78.1	74.2	-	-	71.2	78.4	50.8
Text Perc. [37]	84.1	79.4	67.9	80.5	76.6	65.1	69.7	78.3	-	-	-
CRAFTS [3]	-	-	-	83.1	**82.1**	74.9	-	-	**78.7**	-	-
MANGO*† [36]	85.2	81.1	74.6	**85.4**	80.1	73.9	72.9	83.6	68.9‡	78.9‡	-
YAMTS* [16]	**86.8**	82.4	76.7	85.3	79.8	74.0	-	-	71.1	78.4	-
Ours*	**86.8**	**82.5**	**78.8**	84.7	80.1	**76.3**	**79.9**	**86.2**	76.6	**83.0**	**67.1**

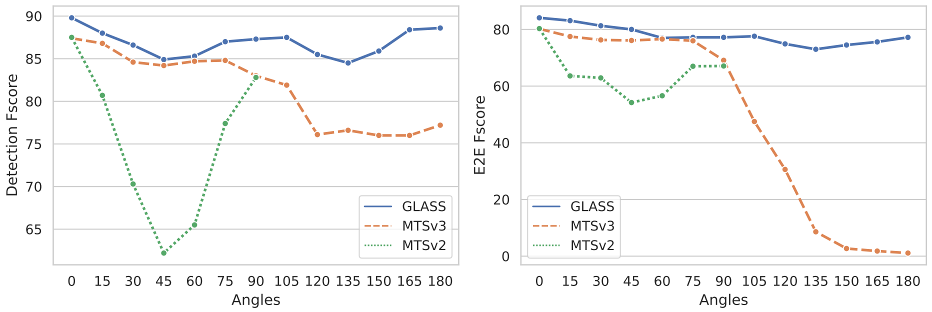

Fig. 4. GLASS contribution for different angles. We measure the performance on the Rotated ICDAR 2013 dataset. Combining GLASS with our novel sine-squared loss improves text detection and recognition across all angles. Notice the 4% detection and 7% recognition gains at angles close to 90°.

To validate our framework performance on oriented text, we show results on the Rotated ICDAR 2013 benchmark [20] in Fig. 4. Our approach with GLASS and the sine-squared loss outperforms previous art on both text detection and recognition across all angles, and especially benefits the detection algorithm on steep angles larger than 60°.

4.4 Incorporating GLASS into other methods

GLASS can be incorporated into any text spotting architecture that uses a feature pooling module, regardless of its specific pooling mechanism. To demonstrate this, we employ GLASS in two common E2E text spotting works, Mask TextSpotter v3 [20] and ABCNet v2 [29]. We note that Mask TextSpotter v3 and ABCNet v2 use different backbones and RPN modules, different detection and

Table 2. GLASS results with Mask TextSpotter v3 (MTSv3) [20] **and ABCNet v2** [29]**.** First and third rows are results reproduced using the official MTSv3 and ABCNet v2 implementations. The second and fourth rows show the effect of incorporating GLASS into MTSv3 and ABCNet v2.

Method	Total-Text		ICDAR 2015	
	E2E Hmean	FPS	E2E Hmean	FPS
MTSv3 [20]	67.5	2.2	68.5	2.6
+ GLASS	69.8 (**+2.3**)	2.0	72.3 (**+3.8**)	2.3
ABCNet v2 [29]	67.6	6.5	-	-
+ GLASS	71.3 (**+3.7**)	6.0	-	-

recognition heads, in addition to other minor differences. Importantly, unlike our method which uses Rotated-RoIAlign pooling, Mask TextSpotter v3 uses an axis-aligned RoIAlign with hard feature masking, and ABCNet v2 applies BezierAlign. Despite the differences between all three architectures, the use of GLASS within both Mask TextSpotter v3 and ABCNet v2 is straightforward and requires minimal changes.

For each method, we use its publicly available source code[3],[4] and train both architectures with and without the GLASS component, following the training procedure published by the respective authors. Results are shown in Table 2. Adding only minor computational overhead, GLASS provides a considerable benefit to the E2E performance of each method. Further discussion on the computational aspects of GLASS is found in the Supplementary Material.

4.5 Ablation study

To evaluate the effectiveness of individual parts in our proposed framework, we conduct ablation studies on the Total-Text and ICDAR15 datasets. In Tables 3 to 6 we report the *end-to-end* F-measure (Hmean) as defined in ICDAR15 [14]. Every model version is pre-trained and fine-tuned as an end-to-end system independently for every experiment.

The **baseline** architecture consists of a Mask R-CNN detection branch with a Rotated-RoIAlign component and our novel rotated box regression loss, as described in Sec. 3.2. The recognizer is set as described in Supplementary Material and receives as input only the *global* features from the shared backbone, as described in Sec. 3.1 and shown in Fig. 1b. The recognition head remains unchanged for all experiments, with the sole difference being the input features selected for it.

Contribution of GLASS to end-to-end performance The effect of GLASS on overall performance is presented in Table 3. Different feature map and fusion

[3] https://github.com/MhLiao/MaskTextSpotterV3
[4] https://github.com/aim-uofa/AdelaiDet

Table 3. Ablation study - Fusion. "Global" and "Local" columns stands for the use of image-level (Global) and cropped (Local) features. "Fusion" column compares two different fusion operations where both features are used: a simple channel-wise concatenation and our fusion method. All rows use the Rotated Mask R-CNN detector and recognition head described in Sec. 3.2 and Sec. 3.3. "FPS" column states the average latency in frames-per-second measured for Total-Text. We measure End-to-End Hmean on Total-Text (TT) and ICDAR15 (IC15).

	Global	Local	Fusion	TT	IC15	FPS
Baseline	✓		–	72.6	69.1	3.0
Baseline + Local		✓	–	69.4 (↓ 3.2)	64.5 (↓ 4.6)	2.8
Baseline + Global-Local	✓	✓	Concat.	75.0 (↑ 2.4)	72.6 (↑ 3.5)	2.7
Baseline + GLASS	✓	✓	Ours	75.7 (↑ 3.1)	73.7 (↑ 4.6)	2.7

Table 4. Ablation study - Orientation loss. We use Total-Text and ICDAR15 to compare the two losses for the rotated box angle: (a) the commonly used L_1 loss and (b) our novel sine-squared loss from Eq. (4). Both experiments use GLASS. The metrics R, P, and H denote detection recall, precision, and Hmean respecively, while E2E denoted End-to-End Hmean.

	Total-Text				ICDAR 2015			
	R	P	H	E2E	R	P	H	E2E
GLASS + L_1 loss	**86.3**	88.2	87.2	75.0	83.4	85.0	84.2	73.5
GLASS + $\sin^2$ loss	85.5	**90.8**	**88.1**	**75.7**	**84.5**	**86.9**	**85.7**	**73.7**

combinations are compared, all using the baseline architecture for detection. Replacing the global features with local features as recognition input, shown in the second row of the table, causes a 3.2% and 4.6% regression for Total-Text and ICDAR15 datasets, respectively. We identify the main reason for this regression in a noticeable drop in the detection performance (see in Supplementary Material), and the lack of mutual supervision of the detection-recognition.

A simple channel-wise concatenation of global and local features (row 3) improves the results over the baseline by 2.4% and 3.5% on Total-Text and ICDAR15 datasets. Furthermore, using GLASS provides further boosts for both Total-Text and ICDAR15 datasets, reaching 3.1% and 4.6% over the baseline. Overall, adding the GLASS module leads to considerable gains in the E2E performance, while only increasing latency by roughly 10%.

Orientation robustness analysis We compare our framework with two different orientation losses: our proposed sine-squared loss in Eq. (4) and the commonly used L_1 loss. Both models use GLASS. The results on Total-Text and ICDAR15 datasets are presented in Table 4. Relying on the L_1 loss, instead of our $\sin^2$ loss, leads to a result regression of 0.7% on the Total-Text dataset, which emphasizes arbitrarily rotated text.

Scale robustness analysis We establish our method's contribution in challenging cases in both quantitative and qualitative manners. The relative contribution

Table 5. GLASS contribution across different scales. We measure the performance over 4 scale groups in Total-Text: small, medium, large and extra-large, denoted by S, M, L, XL accordingly (see Fig. 5). The specific sizes were chosen for creating equally sized bins of ground-truth instances. GLASS outperforms the other configurations, especially on the lower and higher end of the scales. The baseline achieves comparable performance to GLASS on medium and large scales.

	S	M	L	XL
Baseline	71.8	77.8	77.7	79.0
Baseline + Local branch	72.7 (↑ 1.9)	77.3 (↓ 0.5)	74.9 (↓ 2.8)	75.5 (↓ 3.5)
Baseline + GLASS	**73.9** (↑ 2.1)	**78.1** (↑ 0.3)	**77.8** (↑ 0.1)	**80.8** (↑ 1.8)

Table 6. Ablation study - Recognition with ground-truth boxes. To isolate the recognizer's performance, *ground-truth* boxes are used, simulating perfect detections. We compare global, local and fused recognizer inputs on Total-Text and ICDAR15.

	GT	Features		Total-Text	ICDAR15
	Detection	Global	Local	Hmean	Hmean
Baseline	✓	✓		75.3	72.8
Baseline + Local branch	✓		✓	74.2 (↓ 1.1)	78.5 (↑ 5.7)
Baseline + GLASS	✓	✓	✓	80.5 (↑ 5.2)	80.0 (↑ 7.2)

of GLASS at different scales is quantified by performing a custom ablation study using the Total-Text dataset. Total-Text contains a variety of challenging texts at different scales and rotations. First, we divide text instances into four groups: small, medium, large and extra-large, denoted by S, M, L and XL accordingly, and illustrated in Fig. 5. The groups are defined as four equally-sized bins w.r.t. the square root area of their ground-truth text polygon, over the entire dataset.

The F-score is measured for each population of prediction and ground-truth polygons independently, shown in Table 5. As expected, most of the gain is achieved over the small and extra-large text groups, compared to the baseline that predicts text solely using the global branch.

A qualitative comparison is shown in Fig. 6, where we picked examples from the Total-Text dataset that contain a mix of small and large scale text. The baseline result, shown in row (a), is under-performing on challenging text instances that are either small, very large or have a steep rotation angle. In row (b), where only the local features are used, there is a notable regression in both detection and recognition accuracy. In row (c) our method is robust to both scale and orientation, and capable of accurately detecting and recognizing text in extreme and challenging scenarios.

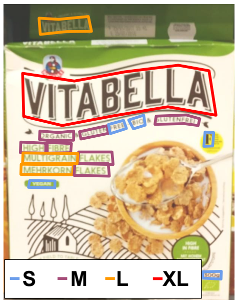

Fig. 5. Illustration of text scale groups.

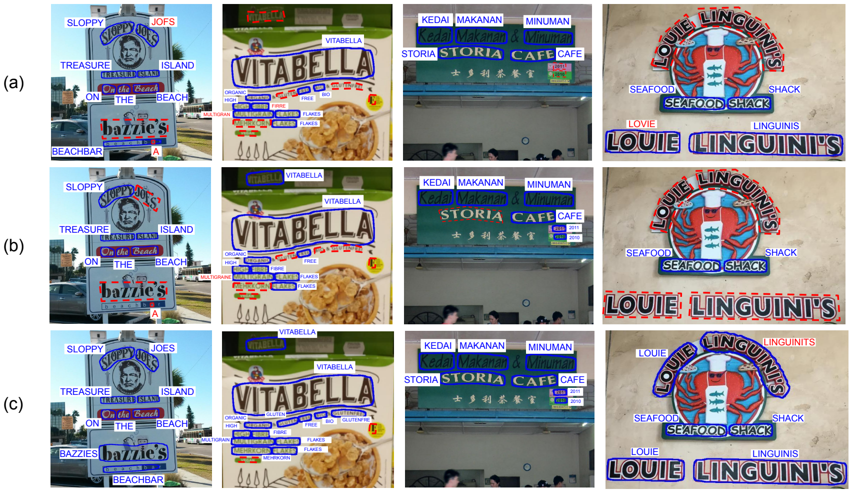

Fig. 6. Qualitative results for Total-Text. Predictions from: **(a)** A standard E2E text spotting framework. **(b)** An E2E framework using only the local features for recognition, and **(c)** Our proposed method with the GLASS component. Blue and red represent correct and incorrect predictions, respectively. GLASS improves recognition, specifically for small and large words, matching the results in Table 5.

Isolating recognition branch performance Lastly, we assess the impact of the GLASS module on the recognition task by injecting the ground-truth rotated boxes as the detection output. By overriding the entire detection branch with *oracle* predictions we are able to isolate the recognizer and compare multiple configurations. The results are presented in Table 6, showing that using the fused features from GLASS contributes to a large increase of 5.2% and 7.2% in the recognition performance on Total-Text and ICDAR15 respectively.

5 Discussion

We propose two extensions for existing text spotting methods. First is combining global and local features for end-to-end text recognition, and a fusion operator enabling that, termed GLASS. The other is an orientation prediction side task, using the orientation-aware sine-squared objective during optimization.

The proposed algorithm combines highly-contextual global features, which also encode each word's surroundings and allows reading it in-context, like humans do, with uniformly scaled and oriented local features, abstracting away scale and rotation. This improves performance for common cases, and even more so for cases of strong scale and rotation.

Extensive experiments over four benchmarks, including the challenging Rotated ICDAR 2013 and the new TextOCR show state-of-the-art results. Ablation studies highlight our contribution over scale and rotation ranges, as well as our method's applicability to other recent text spotting methods.

References

1. Baek, J., Kim, G., Lee, J., Park, S., Han, D., Yun, S., Oh, S.J., Lee, H.: What is wrong with scene text recognition model comparisons? dataset and model analysis. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4715–4723 (2019)
2. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9365–9374 (2019)
3. Baek, Y., Shin, S., Baek, J., Park, S., Lee, J., Nam, D., Lee, H.: Character region attention for text spotting. In: Proceedings of the European Conference on Computer Vision. pp. 504–521. Springer (2020)
4. Bušta, M., Patel, Y., Matas, J.: E2E-MLT - An unconstrained end-to-end method for multi-language scene text. In: Asian conference on computer vision. pp. 127–143. Springer (2018)
5. Cheng, Z., Bai, F., Xu, Y., Zheng, G., Pu, S., Zhou, S.: Focusing attention: Towards accurate text recognition in natural images. Proceedings of the IEEE International Conference on Computer Vision (2017)
6. Ch'ng, C.K., Chan, C.S.: Total-text: A comprehensive dataset for scene text detection and recognition. In: International Conference on Document Analysis and Recognition. vol. 1, pp. 935–942. IEEE (2017)
7. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88**(2), 303–338 (2010)
8. Feng, W., He, W., Yin, F., Zhang, X.Y., Liu, C.L.: TextDragon: An end-to-end framework for arbitrary shaped text spotting. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9076–9085 (2019)
9. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1440–1448 (2015)
10. Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2315–2324 (2016)
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2961–2969 (2017)
12. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. In: Workshop on Deep Learning, Advances in Neural Information Processing Systems (2014)
13. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. In: Advances in Neural Information Processing Systems (2015)
14. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: ICDAR 2015 competition on robust reading. In: International Conference on Document Analysis and Recognition. pp. 1156–1160. IEEE (2015)
15. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: ICDAR 2013 robust reading competition. In: International Conference on Document Analysis and Recognition. pp. 1484–1493. IEEE (2013)
16. Krylov, I., Nosov, S., Sovrasov, V.: Open images v5 text annotation and yet another mask text spotter. arXiv preprint arXiv:2106.12326 (2021)

17. Lee, J., Lee, J., Yang, C., Lee, Y., Lee, J.: Rotated box is back: An accurate box proposal network for scene text detection. In: International Conference on Document Analysis and Recognition. pp. 49–63. Springer (2021)
18. Li, H., Wang, P., Shen, C.: Towards end-to-end text spotting with convolutional recurrent neural networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5238–5246 (2017)
19. Li, H., Wang, P., Shen, C., Zhang, G.: Show, attend and read: A simple and strong baseline for irregular text recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence (2019)
20. Liao, M., Pang, G., Huang, J., Hassner, T., Bai, X.: Mask TextSpotter v3: Segmentation proposal network for robust scene text spotting. In: Proceedings of the European Conference on Computer Vision. pp. 706–722. Springer (2020)
21. Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X.: Real-time scene text detection with differentiable binarization. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, issue 07, pp. 11474–11481 (2020)
22. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2117–2125 (2017)
23. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2117–2125 (2017)
24. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. p. 740–755 (2014)
25. Litman, R., Anschel, O., Tsiper, S., Litman, R., Mazor, S., Manmatha, R.: SCATTER: selective context attentional scene text recognizer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11962–11972 (2020)
26. Liu, W., Chen, C., Wong, K.Y.K.: Char-Net: A character-aware neural network for distorted scene text recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence (2018)
27. Liu, X., Liang, D., Yan, S., Chen, D., Qiao, Y., Yan, J.: FOTS: Fast oriented text spotting with a unified network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5676–5685 (2018)
28. Liu, Y., Chen, H., Shen, C., He, T., Jin, L., Wang, L.: Abcnet: Real-time scene text spotting with adaptive bezier-curve network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9809–9818 (2020)
29. Liu, Y., Shen, C., Jin, L., He, T., Chen, P., Liu, C., Chen, H.: ABCNet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. arXiv preprint arXiv:2105.03620 (2021)
30. Long, S., Yao, C.: Unrealtext: Synthesizing realistic scene text images from the unreal world. arXiv preprint arXiv:2003.10608 (2020)
31. Lu, N., Yu, W., Qi, X., Chen, Y., Gong, P., Xiao, R., Bai, X.: Master: Multi-aspect non-local network for scene text recognition. Pattern Recognition **117**, 107980 (2021)
32. Luo, W., Li, Y., Urtasun, R., Zemel, R.: Understanding the effective receptive field in deep convolutional neural networks. Advances in neural information processing systems **29** (2016)
33. Lyu, P., Liao, M., Yao, C., Wu, W., Bai, X.: Mask TextSpotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In: Proceedings of the European Conference on Computer Vision. pp. 67–83 (2018)

34. Ma, J., Shao, W., Ye, H., Wang, L., Wang, H., Zheng, Y., Xue, X.: Arbitrary-oriented scene text detection via rotation proposals. IEEE Transactions on Multimedia **20**(11), 3111–3122 (2018)
35. Qian, W., Yang, X., Peng, S., Yan, J., Guo, Y.: Learning modulated loss for rotated object detection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, issue 3, pp. 2458–2466 (2021)
36. Qiao, L., Chen, Y., Cheng, Z., Xu, Y., Niu, Y., Pu, S., Wu, F.: MANGO: A mask attention guided one-stage scene text spotter. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, issue 3, pp. 2467–2476 (2021)
37. Qiao, L., Tang, S., Cheng, Z., Xu, Y., Niu, Y., Pu, S., Wu, F.: Text perceptron: Towards end-to-end arbitrary-shaped text spotting. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, issue 07, pp. 11899–11907 (2020)
38. Qin, S., Bissacco, A., Raptis, M., Fujii, Y., Xiao, Y.: Towards unconstrained end-to-end text spotting. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4704–4714 (2019)
39. Qin, X., Zhou, Y., Guo, Y., Wu, D., Tian, Z., Jiang, N., Wang, H., Wang, W.: Mask is all you need: Rethinking mask R-CNN for dense and arbitrary-shaped scene text detection. arXiv preprint arXiv:2109.03426 (2021)
40. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
41. Shi, B., Yang, M., Wang, X., Lyu, P., Yao, C., Bai, X.: ASTER: An attentional scene text recognizer with flexible rectification. IEEE Transactions on Pattern Analysis and Machine Intelligence **41**(9), 2035–2048 (2018)
42. Singh, A., Pang, G., Toh, M., Huang, J., Galuba, W., Hassner, T.: Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8802–8812 (2021)
43. Tan, M., Pang, R., Le, Q.V.: EfficientDet: Scalable and efficient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
44. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., Xiao, B.: Deep high-resolution representation learning for visual recognition. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
45. Wang, P., Li, H., Shen, C.: Towards end-to-end text spotting in natural scenes. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
46. Worrall, D.E., Garbin, S.J., Turmukhambetov, D., Brostow, G.J.: Harmonic networks: Deep translation and rotation equivariance. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition pp. 7168–7177 (2017)
47. Zhou, X., Yao, C., Wen, H., Wang, Y., Zhou, S., He, W., Liang, J.: East: an efficient and accurate scene text detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5551–5560 (2017)
48. Zhu, Y., Ma, C., Du, J.: Rotated cascade R-CNN: A shape robust detector with coordinate regression. Pattern recognition **96**, 106964 (2019)

Supplementary Material

In the supplementary material, we share further details and analysis for our work. In Sec. 6, we provide estimates for the latency and computations costs added when using GLASS. In Sec. 7, we discuss our recognition head selection and further implementation details of our main method. Finally, in Sec. 8, we explore GLASS's effect over detection performance, share results for the TextOCR validation set, present failure cases and further explore GLASS's contribution across different scales.

6 GLASS Computational Cost

In Table 7 we report the number of parameters in our model's detection branch, GLASS component and the recognition branch. During inference, GLASS preforms two main operations: (a) Sampling high-resolution crops and extracting their local features, and (b) Fusing the global features of each detected bounding box with its local feature counterparts.

To minimize the added computational cost, we utilize a light-weight ResNet34 backbone [5] for extracting the local features. As described in Sec. 3.1, we use a block-based attention for fusing the local and global feature maps. This alleviates much of the fusion computational cost, while still benefiting overall performance, as demonstrated in our experiments. Overall, the addition of GLASS leads to an inference time increase of roughly 10%. The latency increment is similar when incorporating GLASS into Mask TextSpotter v3 [20] and ABCNet v2 [29] frameworks, shown in Table 7. GLASS has no effect on the computational cost of the detector, including its mask branch, and the recognizer heads.

7 Implementation Details

7.1 Recognition Head

We follow recent art [29], and design a light-weight recognition head. The recognition head consists of 2 convolution layers, a two-layer BiLSTM encoder, and an attention decoder [1]. The input to the first block of the recognizer, the encoder, is the expressive feature map $\mathbf{z}^{\text{fused}}$ computed by GLASS component. The loss used to train the recognition head is the Negative Log-Likelihood (NLL) as in [5], denoted by $\mathcal{L}_{\text{rec}}$.

We note that our GLASS component is modular, and benefits a variety of additional end-to-end recognition branches, including those found in Mask Text Spotter v3 [20], and ABCNet v2 [29].

Table 7. Model's number of parameters. "FPS" column states the frames-per-second measured for Total-Text dataset.

Method	Detection Branch	Recognition Head	GLASS ResNet34	GLASS Fusion	Total # Parameters	FPS
Baseline	48.8M	3.2M	-	-	52M	2.7
+ GLASS	48.8M	3.2M	10.5M	1.5M	64M	3.0
MTSv3 [20]	41.3M	4.0M	-	-	45.3M	2.3
+ GLASS	41.3M	4.0M	10.5M	1.5M	57.3M	2.6
ABCNet v2 [29]	44.7M	3.0M	-	-	47.7M	6.0
+ GLASS	44.7M	3.0M	10.5M	1.5M	59.7M	6.5

7.2 Training and Optimization

We use a ResNet50 backbone with ImageNet [40] pre-trained weights. Data augmentation includes multiple scales, pixel-level augmentations (color jitter), affine transformations and image cropping. In all of our experiments, we set GLASS feature parameters to $C = 256$, $H = 8$, $W = 32$ and $k = 8$. The recognition head is trained to classify 96 characters, which covers alphabets, numbers, and special characters.

Recall that the overall loss function $\mathcal{L}$ used for the E2E supervised training is given by

$$\mathcal{L} = \mathcal{L}_{\text{rbox}} + \lambda_1 \mathcal{L}_{\text{mask}} + \lambda_2 \mathcal{L}_{\text{rec}} \ , \tag{6}$$

where the mask loss $\mathcal{L}_{\text{mask}}$ is identical to Mask RCNN [11] and $\mathcal{L}_{\text{rec}}$ is the recognition loss, described in Sec. 7.1. In our experiments, we set $\lambda_1 = 0.005$, and observe that higher values hurt the E2E text recognition performance. Also, we empirically set $\lambda_2 = 2$, and the $\mathcal{L}_{\text{rbox}}$ constants are chosen as $\alpha_1 = \alpha_2 = \alpha_\theta = 10$ and $\alpha_3 = \alpha_4 = 5$. During inference, we resize the longer side of the input image to 1600 for ICDAR15 and ICDAR13 datasets and the shorter side to 1000 for Total-Text and TextOCR datasets.

8 Further Analysis

8.1 Detection Performance

In this section, we discuss the contribution of GLASS to detection performance. The recognition head remains unchanged throughout all experiments. The sole difference are the input features used for recognition during both training and inference. Here we present in Table 8, that contains complementary data to Table 3.

The first observation from Table 8, is that we observe a steep performance drop when using only the local branch. This drop is expected, since when using only the local branch for recognition, the detection backbone can not leverage

Table 8. Ablation study - Detection. This table complements paper Table 3 with detection results. "Fusion Type" stands the fusion operator used when both feature types are included: channel-wise concatenation and our fusion method.

	Features		Fusion	Total-Text			ICDAR15		
	Global	Local	Type	R	P	H	R	P	H
Baseline	✓			85.8	90.3	88.0	83.2	87.9	85.5
Baseline + Local		✓		85.4	88.3	86.8	66.6	81.4	73.3
Baseline + Global-Local Concat.	✓	✓	Concat	**87.5**	89.3	**88.4**	81.1	**89.6**	85.2
Baseline + GLASS	✓	✓	Ours	85.5	**90.8**	88.1	**84.5**	86.9	**85.7**

Table 9. Results on the TextOCR validation and test datasets. R, P, and H refer to recall, precision and H-mean. No lexicon is used. Our method with GLASS module outperforms Mask TextSpotter v3 on the test set, noting both approaches were optimized on similar data, including TextOCR train data [42]. On TextOCR validation dataset, our method with the GLASS component surpasses the baseline by a large margin for end-to-end recognition and word spotting metrics.

Method	Validation set					Test set	
	Detection			Word Spotting	End-to-End	Word Spotting	End-to-End
	R	P	H				
MTSv3 [20]	-	-	-	-	-	-	50.8
Baseline	**73.3**	82.5	**77.6**	60.6	55.6	58.0	56.8
GLASS	71.5	**84.3**	77.4	**71.3**	**64.8**	**70.4**	**67.1**

the supervised recognition as an auxiliary task, which was previously shown to improve detection results in [38,45].

In the same vein, we see marginal differences between the baseline, which attempts to recognize text only using the global features, and the other approaches that combine global and local information. This is mainly because the back-propagation path and additional supervision to the detection backbone, only occurs via the global feature branch, as can be seen in Fig. 2. A simple channel-wise concatenation of global and local features or using GLASS module, rows 3 and 4 respectively, shows only minor gains for detection compared to the baseline.

8.2 TextOCR Results

In Table 9, we report end-to-end text recognition and detection results on TextOCR validation dataset and end-to-end results for TextOCR test dataset. Test set results are repeated for brevity. TextOCR validation and test datasets contain a similar number of images and text instances. Incorporating GLASS module into the baseline architecture increases the end-to-end results by a large margin for both validation and test datasets.

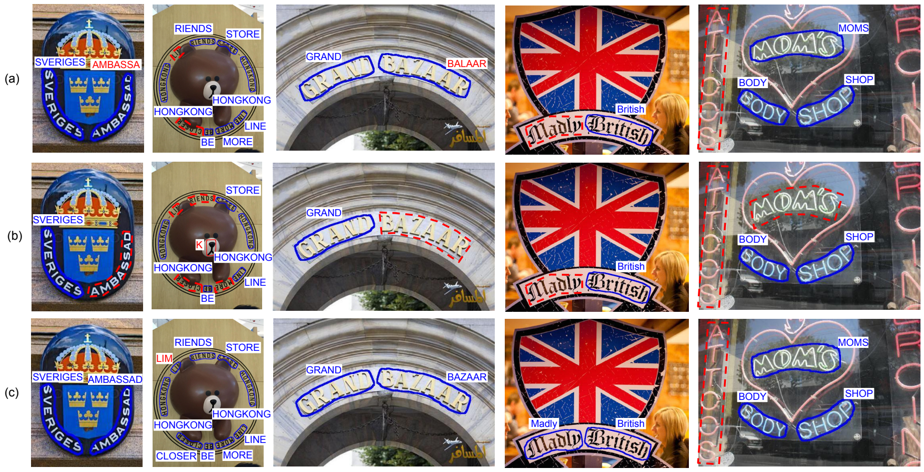

Fig. 7. Qualitative results on the Total-Text dataset. (a) Predictions of the baseline experiment, a standard E2E text spotting framework. **(b)** Predictions of an E2E framework where only the local (crop-level) features are used by the recognizer, and **(c)** Predictions of our proposed configuration with the GLASS component. Polygons and transcriptions in blue represent correct predictions, and red represents wrong predictions. We observe that an E2E system trained with GLASS is capable of detecting with a higher word recall, and higher recognition accuracy. We recommend enlarging the digital version.

8.3 Qualitative Results

Additional qualitative results from Total-Text are presented in Fig. 7. Our method, Fig. 7c, shows high detection and recognition accuracy on curved, rotated, upside down and occluded text and on a variety of fonts. It can be seen in Fig. 7b that using only the crop-level features in the recognizer leads to a regression in both detection and recognition accuracy. In Fig. 7a we present the results of our baseline. Although the text detection accuracy is qualitatively high, the lack of crop-level features leads to worse recognition performance compared to our model which uses both local and global features.

8.4 In-Depth Scale Performance Analysis

We further explore the contribution of GLASS w.r.t. different sized words in an image, and present the performance in relation to additional intrinsic scale properties for word instances in Fig. 8. The left column in the figure presents the histogram of all word instances and their distribution for both area, width and height. The middle column shows the E2E Fscore over four quantiles marked with S, M, L and XL, denoting the relative size groups for the text instances.

We note that regardless of the scale dimension on which we perform an analysis, either area, width or height, we observe the same trend. GLASS out-

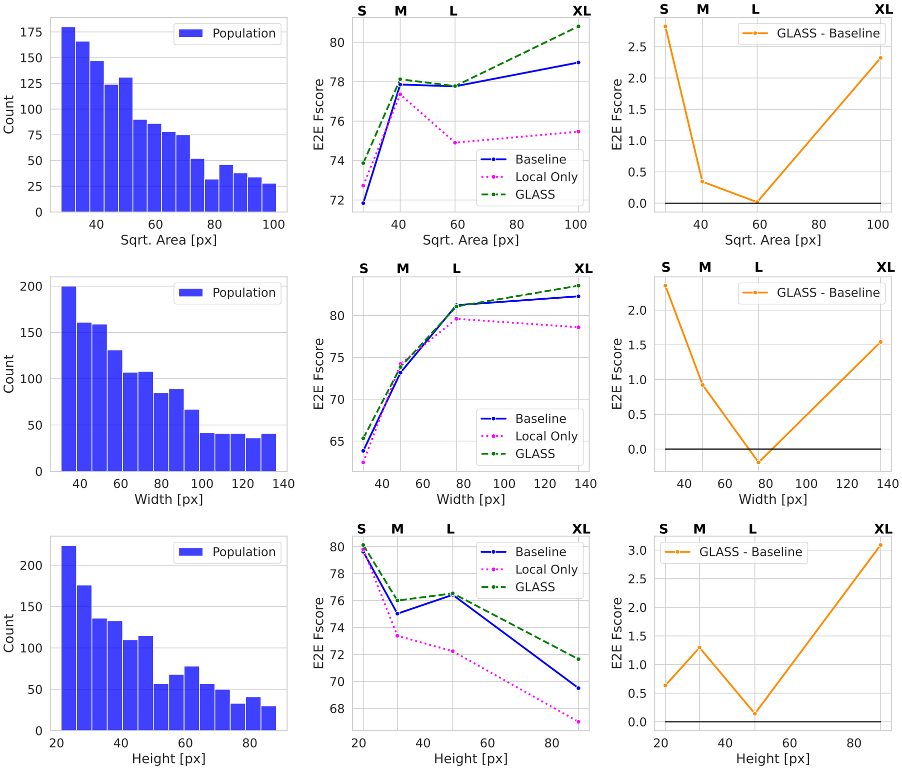

Fig. 8. In-depth performance analysis for different scales. Different text instances are over four different size groups S, M, L and XL. We analyze end-to-end recognition performance for 3 different text scale dimensions: (a) Polygon area, (b) Rotated bounding box width and (c) Rotated bounding box height. For all different scale properties, GLASS increases performance over the baseline, especially on the extremities of small and large text.

performs the baseline by up to 3 percentage points on the edges, measuring the performance on either small or large text instances.

8.5 Failure Cases

We show failure cases of our model with GLASS component and our novel orientation loss in Fig. 9. We stress that our model fails to detect text instances with a large space among the characters. It may be a result of our anchor based Mask R-CNN detection branch. Additionally, we see that our model struggles to detect and recognize text with irregular font. It may be resolved by training the model on a larger dataset.

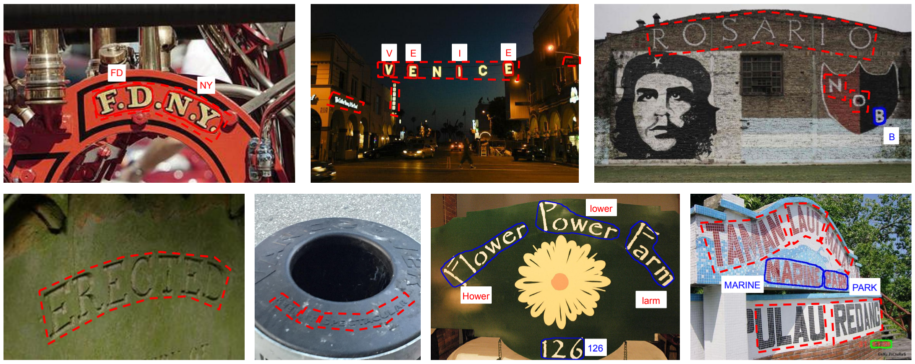

Fig. 9. Failure cases of our model with GLASS component on the Total-Text dataset. In the upper images, we see that our model fails to detect text instances with a large space among the characters. In the second row of images, our model struggles to detect and recognize text with irregular font.