GLASS: Global to Local Attention for Scene-Text Spotting - Supplementary Material

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

In the supplementary material, we share further details and analysis for our work. In Sec. 1, we provide estimates for the latency and computations costs added when using GLASS. In Sec. 2, we discuss our recognition head selection and further implementation details of our main method. Finally, in Sec. 3, 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.

1 GLASS Computational Cost

In Table 1 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 [2] for extracting the local features. As described in the manuscript, 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 [4] and ABCNet v2 [5] frameworks, shown in Table 1. GLASS has no effect on the computational cost of the detector, including its mask branch, and the recognizer heads.

2 Implementation Details

2.1 Recognition Head

We follow recent art [5], 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

* Equal contribution. For correspondence: tsiper@amazon.com

Table 1. 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 [4]	41.3M	4.0M	-	-	45.3M	2.3
+ GLASS	41.3M	4.0M	10.5M	1.5M	57.3M	2.6
ABCNet v2 [5]	44.7M	3.0M	-	-	47.7M	6.0
+ GLASS	44.7M	3.0M	10.5M	1.5M	59.7M	6.5

used to train the recognition head is the Negative Log-Likelihood (NLL) as in [2], 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 [4], and ABCNet v2 [5].

2.2 Training and Optimization

We use a ResNet50 backbone with ImageNet [7] 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

$$\mathcal{L} = \mathcal{L}_{\text{rbox}} + \lambda_1 \mathcal{L}_{\text{mask}} + \lambda_2 \mathcal{L}_{\text{rec}} \,, \tag{1}$$

where the mask loss $\mathcal{L}_{\text{mask}}$ is identical to Mask RCNN [3] and $\mathcal{L}_{\text{rec}}$ is the recognition loss, described in Sec. 2.1. 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.

3 Further Analysis

3.1 Detection Performance

In this section, we discuss the contribution of GLASS to detection performance. The recognition head remains unchanged throughout all experiments. The sole

Table 2. 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 3. 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 [8]. 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 [4]	-	-	-	-	-	-	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**

difference are the input features used for recognition during both training and inference. Here we present in Table 2, that contains complementary data to Table 3 of the main manuscript.

The first observation from Table 2, 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 the supervised recognition as an auxiliary task, which was previously shown to improve detection results in [6,9].

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 in the main manuscript. 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.

Fig. 1. 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.

3.2 TextOCR Results

In Table 3, 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.

3.3 Qualitative Results

Additional qualitative results from Total-Text are presented in Fig. 1. Our method, Fig. 1c, 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. 1b that using only the crop-level features in the recognizer leads to a regression in both detection and recognition accuracy. In Fig. 1a 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.

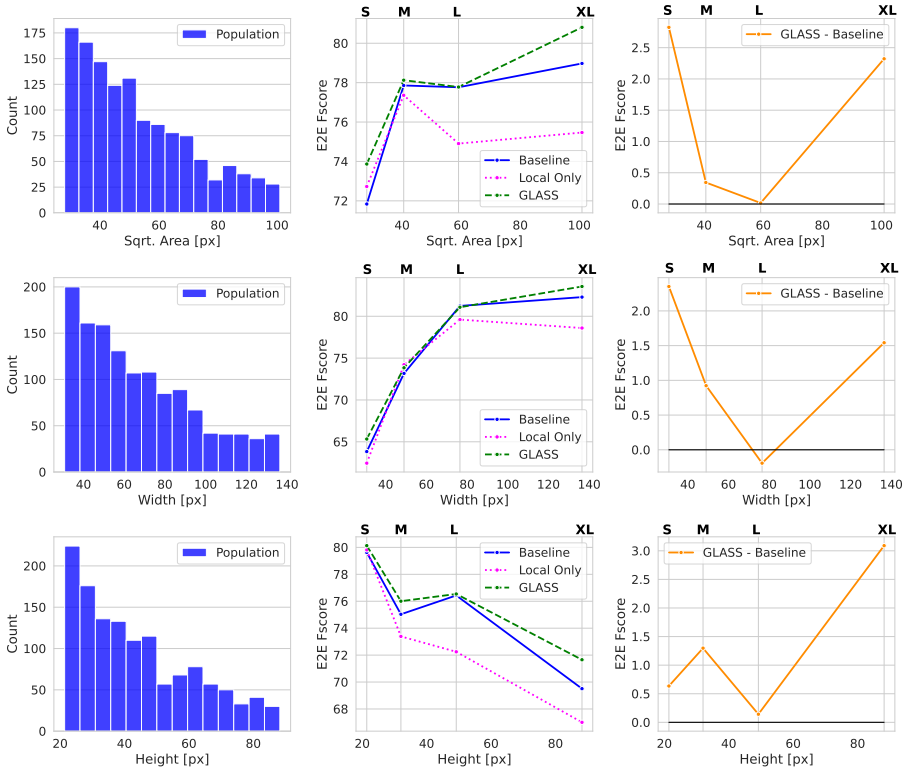

Fig. 2. In-depth performance analysis for different scales. Different text instances are over four different size groups S, M, L and XL. We analyze end-to-end recognition performance for 3 different text scale dimensions: (a) Polygon area, (b) Rotated bounding box width and (c) Rotated bounding box height. For all different scale properties, GLASS increases performance over the baseline, especially on the extremities of small and large text.

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

Fig. 3. 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.

3.5 Failure Cases

We show failure cases of our model with GLASS component and our novel orientation loss in Fig. 3. 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.

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. 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)
3. 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)
4. 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)
5. 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)
6. 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)
7. 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)
8. 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)
9. 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)