What's Wrong with Synthetic Data for Scene Text Recognition? A Strong Synthetic Engine with Diverse Simulations and Self-Evolution

Supplementary Material

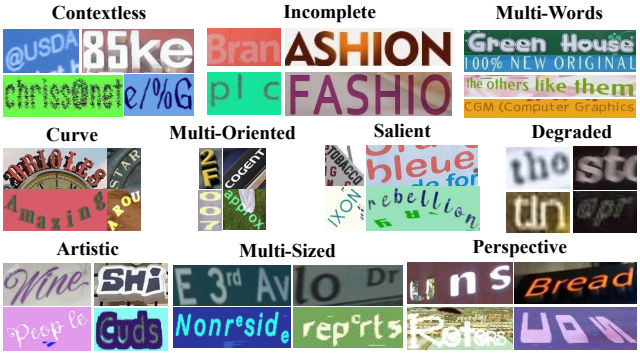

Figure 5. Real text samples (the top line in each subset) and data generated by UnionST (the bottom line) grouped by subsets according to their challenges. Beyond the subsets identified in Union14M [19], we introduce three additional ones: *Multi-Sized* (words with varying sizes, including subscripts and superscripts), *Perspective* (variations in viewpoint), and *Degraded* (blur or low resolution caused by camera shake or small text size).

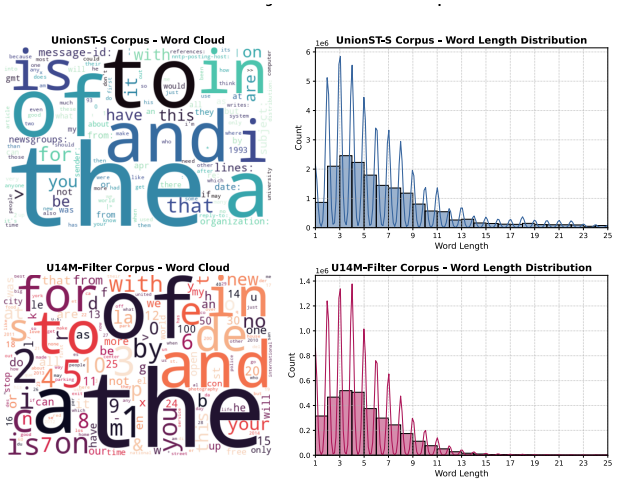

Figure 6. Word clouds (left) and word length distributions (right) for the synthetic corpus (top) and the real (bottom).

6. UnionST Details

6.1. Corpus

The details of the previously mentioned corpus construction are as follows:

- **Common:** We augment the MJ [18]&ST [16] corpora, applying case transformations (original, lowercase, uppercase, capitalized) to obtain 264K and 13M samples, respectively.
- **Contextless**: Random character sequences (length 2–25, 10K per length, totaling 240K) are generated from a character set consisting of 94 commonly used alphabetic and symbolic characters.
- **Incomplete**: We create incomplete words by randomly removing initial, terminal, or internal characters from words in the MJ corpus, yielding 264K samples.
- **Multi-Words**: We collect 400K common phrases and concatenated multi-word expressions. Additionally, we extract substrings of varying lengths (1–25, 120K per length, totaling 3M) from the ST newspaper corpus.

Fig. 6 shows that the UnionST-S corpus demonstrates a word length distribution for short texts that closely aligns with the real dataset, while also providing a higher proportion of longer texts. Fig. 8 further shows that the UnionST-S corpus places more emphasis on certain rare or special characters. And Fig. 7 illustrates the word distribution among different corpora in UnionST. It can be seen that the pseudo-

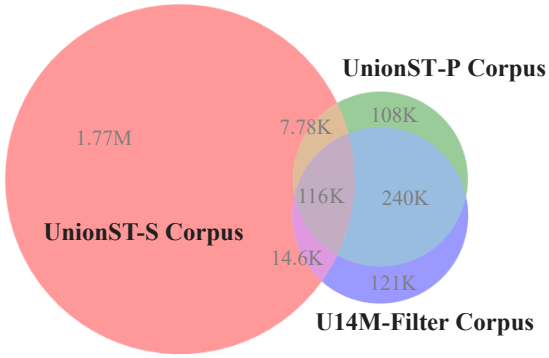

Figure 7. The Venn diagram comparing words from three different corpora reveals a substantial disparity between the synthesized corpus and the real corpus.

labeling approach enables us to obtain results at scale that closely match those of real corpora, leaving only 121K words in UnionST-SP uncovered. At the same time, this approach provides a much larger corpus overall.

6.2. Font

We curate 113,788 font files from publicly available open-source repositories. For ablation studies, we use a filtered subset of 3,092 fonts from the Google Fonts collection for comparison. During the collection process, we pay special

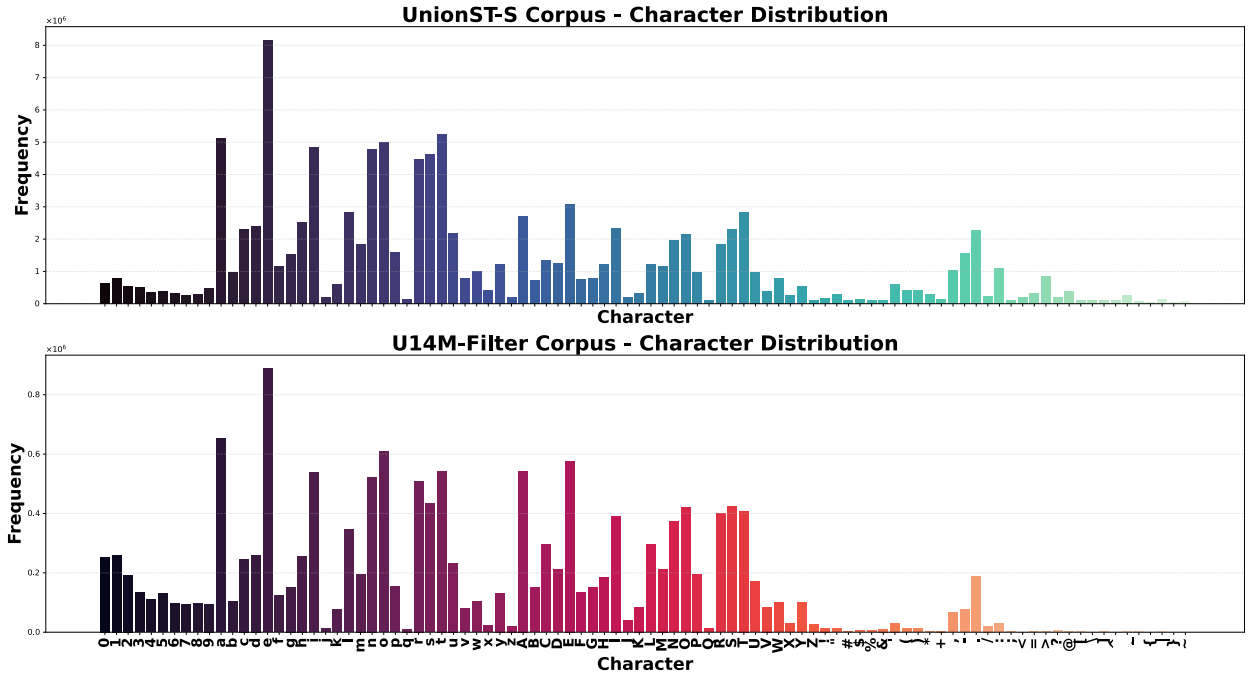

Figure 8. Character distribution of the UnionST-S Corpus (top) and the U14M-Filter Corpus (bottom).

attention to the license terms, ensuring that all fonts are clearly labeled as free for commercial use at their source websites or repositories, and comply with the corresponding open-source licenses (such as SIL Open Font License, Apache License, etc.).

6.3. Other Resources

Other details not mentioned are basically consistent with ST [16] and SynthTIGER [48], such as the color mapping table, background images, and the filtering mechanism after generating images, etc.

6.4. Other Algorithm Improvements

Real images often contain distracting non-target text as mid-ground elements. Following SynthTIGER's strategy [48], we also add non-salient text as background clutter to focus on the main salient text. However, we find that SynthTIGER's implementation has a problem: when handling foreground and mid-ground text, it often fails to account for the differences in their respective sizes. This results in simulated mid-ground text that lacks diversity. As shown in Alg. 2, the underlined part indicates what we add.

6.5. Dataset Properties

Regarding image types, we store all images in the JPEG format and write them into the lmdb files that are commonly used in STR. Fig. 9 shows the distribution of tags, such as modern, display, handwriting, and script, demonstrating the comprehensiveness and diversity of our font coverage.

Algorithm 2 Improved Mid-ground Text Blending

Require: Foreground text F, Mid-ground text M, Background image I_{bg}
Ensure: Composite image I

placement$_M$, placement$_F$ $\leftarrow$ Compute(F, M, I_{bg})
$I_{crop} \leftarrow$ Crop(I_{bg})
$I_{mid} \leftarrow$ Blend$(I_{crop}, M,$ placement$_M)$
$I_{mid} \leftarrow$ EraseOverlap$(I_{mid}, F,$ placement$_F)$
$I \leftarrow$ Blend$(I_{mid}, F,$ placement$_F)$
return I

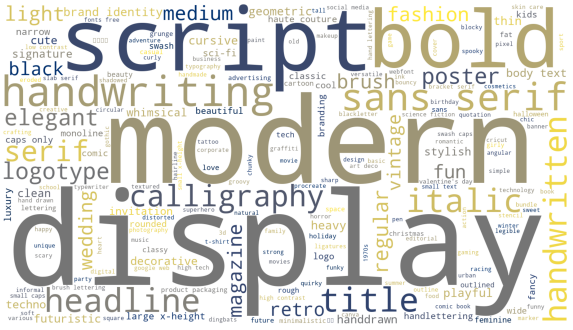

Figure 9. Word cloud of font tags uesd in UnionST.

6.6. Other Variants

If real labels are available, we can naturally use them as the labels for the synthetic data. Therefore, we use the text labels from U14M-Filter [8, 19] as the corpus to construct UnionST-R (also 5M). This allows us to evaluate the performance of UnionST-P.

6.7. Runtime and Cost Efficiency

For UnionST data synthesis, we employ a server equipped with an Intel Xeon Platinum 8255C CPU (96 cores, 375 GB RAM). With 16 worker processes, generating 1 million samples requires 49,040 seconds, corresponding to a throughput of approximately 20.39 samples per second. Compared to deep generative methods, which often rely heavily on GPU resources, UnionST offers significant advantages in both cost and computational efficiency. The CPU cost for UnionST is \$0.2 per hour, so generating 5 million samples incurs a total cost of only \$13.62. For reference, TextSSR reports that a single RTX-3090 GPU can generate 6.15K images per card-hour, meaning that producing 5 million samples requires 813 card-hours and costs \$325.2 at \$0.4 per card-hour. Closed-source Nano Banana charges \$0.039 per image, resulting in a \$195,000 budget for producing 5 million samples. Additionally, the quoted price for manual annotation is \$0.06 per image, meaning annotating 5 million images would cost \$300,000. Our SEL framework reduces annotation costs to \$27,000 and can provide pseudo-labels to accelerate annotation efficiency. Overall, UnionST achieves far lower costs, demonstrating clear resource-efficient characteristics and suitability for large-scale synthetic data generation.

7. Model Details

Encoder: SVTRv2 [8] begins by partitioning the input image into patches and projects it into a high-dimensional embedding space. It then consists of three stages, each comprising Conv-Blocks and Mixing-Blocks (which use local convolution and global self-attention to capture both local details and global context). Given an input image $\mathbf{I} \in R^{H\times W\times 3}$, the encoder produces a visual feature sequence as follows:

$$\mathbf{F}_E = \mathrm{Encoder}(\mathbf{I}), \tag{4}$$

where $\mathbf{F}_E \in R^{L\times C}$, with L representing the sequence length and C the channel dimension.

Decoder: The AR decoder takes the encoder output $\mathbf{F}_E$ and the embedded target character sequence $\mathbf{T}_{1:t-1}$ (with positional encoding), and models the sequence using two layers of multi-head self-attention and cross-attention:

$$\mathbf{H}_t = \mathrm{Decoder}(\mathbf{F}_E, \mathrm{Embed}(\mathbf{T}_{1:t-1}) + \mathrm{PE}_{1:t-1}) \tag{5}$$

where PE denotes sinusoidal positional encoding. The output at each character step t is computed as:

$$P(y_t|y_{<t}, \mathbf{F}_E) = \mathrm{Softmax}(\mathbf{H}_t) \tag{6}$$

During training, we apply teacher forcing to optimize the cross-entropy loss:

$$\mathcal{L}_{\mathrm{CE}} = -\sum_{t=1}^{T} \log P(y_t^*|y_{<t}^*, \mathbf{F}_E), \tag{7}$$

where y_t^* is the ground-truth character at position t. Inference is performed autoregressively.

8. SEL Details

SEL aims to progressively reduce the domain gap between synthetic training data and real-world text by coupling pseudo-label driven corpus construction with iterative self-training. We first train an initial recognizer $\mathcal{M}_a$ on UnionST-S and apply it to a large unlabeled real set $\mathcal{D}_U$ to obtain predictions. The predicted strings $\hat{\mathcal{Y}}_U$ are then used as a target corpus to re-synthesize a new synthetic dataset UnionST-P via the same UnionST rendering engine, yielding UnionST-SP when combined with UnionST-S. Retraining/fine-tuning on UnionST-SP produces a better in-domain model $\mathcal{M}_b$. Starting from $\mathcal{M}_b$, we further perform ISR: at each round t, $\mathcal{M}_{t-1}$ is used to pseudo-label $\mathcal{D}_U$, high-confidence samples (above threshold τ, optionally with simple validity checks such as charset/length constraints) are added as $\mathcal{D}_P^{(t)}$ for fine-tuning to obtain $\mathcal{M}_t$, while the remaining low-confidence subset $\mathcal{D}_U^{\mathrm{low}}$ is kept for later manual annotation since it concentrates hard cases (blur, extreme perspective, rare fonts, occlusion, and low contrast). After T rounds, we annotate only $\mathcal{D}_U^{\mathrm{low}}$ to form $\mathcal{D}_L^{\mathrm{hard}}$ and conduct a final fine-tuning step. In the qualitative results in Fig. 10, we provide intermediate examples across rounds (model prediction and the selected status), illustrating how SEL gradually improves recognition on difficult samples, thereby enhancing both training diversity and target-domain robustness.

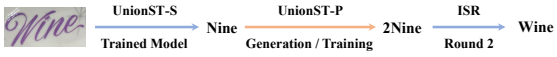

Figure 10. Qualitative examples of intermediate SEL outputs.

9. Training Details

Training schedules are adjusted according to dataset size. For synthetic data, we use 40 epochs as the default for 10M UnionST-SP samples. For larger datasets, we sample 10M instances per epoch for 40 epochs (e.g., ST-2D uses a sampling ratio of 0.28). For smaller datasets, the number of

Type	Model	Training Data	Common Benchmarks							Union14M-Benchmark							
			IIIT	*SVT*	*IC13*	*IC15*	*SVTP*	*CUTE*	*AVG*	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
CTC	CRNN [37]	U14M-Filter	95.80	91.80	94.60	84.90	83.10	91.00	90.21	48.10	13.00	51.20	62.30	41.40	60.40	68.20	49.24
		UnionST-S	92.90	85.63	91.83	75.37	76.12	82.29	84.02	33.59	64.21	36.33	49.29	16.49	65.78	48.98	44.95
		Δ	-2.90	-6.17	-2.77	-9.53	-6.98	-8.71	-6.19	-14.51	51.21	-14.87	-13.01	-24.91	5.38	-19.22	-4.29
	ViT-CTC	U14M-Filter	97.20	94.13	95.45	86.03	87.75	92.01	92.10	70.03	80.50	58.00	72.53	68.04	73.06	77.56	71.39
		UnionST-S	96.33	93.35	97.20	83.49	88.22	92.71	91.88	68.47	84.59	58.00	71.12	52.24	79.61	65.71	68.53
		Δ	-0.87	-0.78	1.75	-2.54	0.47	0.70	-0.22	-1.56	4.09	0.00	-1.41	-15.80	6.55	-11.85	-2.86
	SVTR [5]	U14M-Filter	98.00	97.10	97.30	88.60	90.70	95.80	94.58	76.20	44.50	67.80	78.70	75.20	77.90	77.80	71.17
		UnionST-S	97.50	94.28	97.78	85.64	90.08	95.49	93.46	74.73	86.56	62.44	74.45	62.67	81.80	68.67	73.05
		Δ	-0.50	-2.82	0.48	-2.96	-0.62	-0.31	-1.12	-1.47	42.06	-5.36	-4.25	-12.53	3.90	-9.13	1.88
	SVTRv2 [8]	U14M-Filter	99.20	98.00	98.70	91.10	93.50	99.00	96.57	90.60	89.00	79.30	86.10	86.20	86.70	85.10	86.14
		UnionST-S	98.20	95.98	98.60	87.85	94.26	97.22	95.35	83.39	91.31	67.11	77.41	70.31	83.25	71.10	77.70
		Δ	-1.00	-2.02	-0.10	-3.25	0.76	-1.78	-1.22	-7.21	2.31	-12.19	-8.69	-15.89	3.45	-14.00	-8.44
PD	ABINet [11]	U14M-Filter	98.50	98.10	97.70	90.10	94.10	96.50	95.83	80.40	69.00	71.70	74.70	77.60	76.80	79.80	75.72
		UnionST-S	97.00	93.35	96.97	85.04	90.39	94.10	92.81	73.04	85.10	58.44	64.31	59.25	77.31	63.98	68.78
		Δ	-1.50	-4.75	-0.73	-5.06	-3.71	-2.40	-3.02	-7.36	16.10	-13.26	-10.39	-18.35	0.51	-15.82	-6.94
	LPV [50]	U14M-Filter	98.60	97.80	98.10	89.80	93.60	97.60	95.93	86.20	78.70	75.80	80.20	82.90	81.60	82.90	81.20
		UnionST-S	97.77	95.98	96.85	86.47	90.85	94.44	93.73	78.44	89.85	65.00	71.63	66.65	81.31	69.25	74.59
		Δ	-0.83	-1.82	-1.25	-3.33	-2.75	-3.16	-2.20	-7.76	11.15	-10.80	-8.57	-16.25	-0.29	-13.65	-6.61
	BUSNet [41]	U14M-Filter	98.30	98.10	97.80	90.20	95.30	96.50	96.06	83.00	82.30	70.80	77.90	78.80	71.20	82.60	78.10
		UnionST-S	97.20	95.52	96.62	85.75	91.94	94.10	93.52	73.62	87.66	63.44	69.58	62.22	79.61	68.70	72.12
		Δ	-1.10	-2.58	-1.18	-4.45	-3.36	-2.40	-2.54	-9.38	5.36	-7.36	-8.32	-16.58	8.41	-13.90	-5.98
	CPPD [7]	U14M-Filter	99.00	97.80	98.20	90.40	94.00	99.00	96.40	86.20	78.70	76.50	82.90	83.50	81.90	83.50	81.91
		UnionST-S	97.17	95.36	96.85	85.15	91.94	95.49	93.66	83.80	90.80	67.11	75.74	70.06	82.04	69.82	77.05
		Δ	-1.83	-2.44	-1.35	-5.25	-2.06	-3.51	-2.74	-2.40	12.10	-9.39	-7.16	-13.44	0.14	-13.68	-4.86
AR	PARSeq [3]	U14M-Filter	98.90	98.10	98.40	90.10	94.30	98.60	96.40	87.60	88.80	76.50	83.40	84.40	84.30	84.90	84.26
		UnionST-S	97.77	94.74	97.32	85.64	92.09	93.75	93.55	73.54	88.31	64.89	72.66	68.48	79.85	70.24	74.00
		Δ	-1.13	-3.36	-1.08	-4.46	-2.21	-4.85	-2.85	-14.06	0.49	-11.61	-10.74	-15.92	4.45	-14.66	-10.26
	MAERec [19]	U14M-Filter	99.20	97.80	98.20	90.40	94.30	98.30	96.36	89.10	87.10	79.00	84.20	86.30	85.90	84.60	85.17
		UnionST-S	98.10	96.91	98.48	88.13	95.97	95.83	95.57	78.94	90.80	68.44	77.02	71.45	83.01	73.29	77.56
		Δ	-1.10	-0.89	0.28	-2.27	1.67	-2.47	-0.79	-10.16	3.70	-10.56	-7.18	-14.85	-2.89	-11.31	-7.61
	OTE [44]	U14M-Filter	98.60	96.60	98.00	90.10	94.00	97.20	95.74	86.00	75.80	74.60	74.70	81.00	65.30	82.30	77.09
		UnionST-S	97.77	94.74	97.32	85.64	92.09	93.75	93.55	73.54	88.31	64.89	72.66	68.48	79.85	70.24	74.00
		Δ	-0.83	-1.86	-0.68	-4.46	-1.91	-3.45	-2.19	-12.46	12.51	-9.71	-2.04	-12.52	14.55	-12.06	-3.09
	SMTR [6]	U14M-Filter	99.00	97.40	98.30	90.10	92.70	97.90	95.90	89.10	87.70	76.80	83.90	84.60	89.30	83.70	85.00
		UnionST-S	97.67	95.52	97.78	86.36	91.01	94.10	93.74	80.87	88.31	65.33	76.89	69.24	84.59	71.60	76.69
		Δ	-1.33	-1.88	-0.52	-3.74	-1.69	-3.80	-2.16	-8.23	0.61	-11.47	-7.01	-15.36	4.71	-12.10	-8.31
	SVTRv2-AR	U14M-Filter	99.03	97.37	98.60	90.56	95.50	98.26	96.56	91.71	94.74	79.44	86.01	86.86	86.29	85.47	87.22
		UnionST-S	98.20	95.98	98.48	88.29	94.11	96.87	95.32	88.99	94.45	73.11	80.62	81.68	87.26	74.87	83.00
		Δ	-0.83	-1.39	-0.12	-2.27	-1.39	-1.39	-1.24	-2.72	-0.29	-6.33	-5.39	-5.18	0.97	-10.60	-4.22

Table 8. Quantitative comparison of different STR models trained on real and synthetic datasets. Δ denotes the difference between the results obtained on synthetic data and those on real data. Red indicates a negative value, while blue indicates a positive value.

epochs is increased proportionally (e.g., 58 epochs for ST). For real data fine-tuning with fewer than 0.1M samples, we train for 100 epochs; for larger real datasets, we use 20 epochs, consistent with training from scratch. For the confidence threshold τ of ISR, it is set to 0.9. The initial learning rate is set to 6.5×10^{-4}, while fine-tuning uses 5×10^{-5}. UnionST synthetic data training employs online resampling augmentation (0.1–1×) and simulates compression loss with a probability of 0.2. Images with a height greater than 1.5 times their width are rotated 90 degrees counterclockwise.

Other settings align with SVTRv2 [8]: We use the AdamW optimizer [27] with a weight decay of 0.05. The batch size is set to 256. One cycle LR scheduler [28] with 1.5 epochs linear warm-up is used in all epochs, and the maximum text length is set to 25 during training. The character set size is set to 94, including numbers, uppercase and lowercase letters, and common symbols. All models are trained on 8 RTX V100 GPUs. For synthetic-real data mixing, training is first conducted on synthetic data, followed by fine-tuning on real data.

10. More Results

10.1. Experiments on Varying Text Lengths

Fig. 11 further examines the relationship between label length and recognition accuracy. Across all training datasets, models achieve higher accuracy on shorter labels, with performance declining as label length increases. This trend reflects the increased difficulty of recognizing longer

Training Data	Volume	Common Benchmarks							Union14M-Benchmark							
		IIIT	*SVT*	*IC13*	*IC15*	*SVTP*	*CUTE*	*AVG*	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
UnionST-S	5.00M	98.20	95.98	98.48	88.29	94.11	96.87	95.32	88.99	94.45	73.11	80.62	**81.68**	87.26	74.87	83.00
UnionST-S	10.0M	98.66	95.98	98.71	88.13	94.88	96.87	95.54	88.13	93.42	73.67	82.80	79.66	87.74	77.62	83.29
UnionST-P	5.00M	98.70	97.22	98.60	89.23	95.97	97.57	96.21	89.45	95.03	78.22	83.83	81.49	83.50	78.58	84.30
UnionST-R	5.00M	98.63	96.91	98.60	89.23	95.35	**98.61**	96.22	89.28	95.47	77.33	85.62	81.43	84.95	79.39	84.78
UnionST-SP	10.0M	98.60	97.37	**99.07**	89.29	94.88	97.22	96.07	90.64	**95.69**	75.67	84.98	80.16	**87.99**	78.92	84.86
UnionST-SR	10.0M	98.53	**97.68**	98.60	**89.67**	96.28	96.87	96.27	90.48	95.03	76.67	85.37	80.99	87.50	78.98	85.00
UnionST-PR	10.0M	**98.87**	97.06	98.72	89.40	**96.43**	**98.61**	**96.51**	**90.73**	95.54	**79.00**	**85.88**	81.49	87.38	**79.56**	**85.65**
UnionST-S + R	10.0M + 3.22M	99.37	98.30	**99.30**	92.05	**97.83**	**99.99**	97.81	94.93	97.08	85.11	88.45	90.52	91.99	87.09	90.74
UnionST-SP + R	10.0M + 3.22M	**99.47**	**98.92**	**99.30**	**92.44**	96.90	**99.99**	97.84	**95.42**	97.22	**86.67**	**89.60**	**91.09**	91.99	**87.74**	**91.39**
UnionST-SR + R	10.0M + 3.22M	**99.47**	98.76	**99.30**	92.32	**97.83**	**99.99**	**97.95**	95.05	97.22	86.22	88.83	90.65	**92.48**	87.36	91.12
UnionST-PR + R	10.0M + 3.22M	**99.47**	**98.92**	99.07	**92.44**	97.36	**99.99**	97.88	95.14	**97.30**	86.00	89.22	90.78	91.38	87.64	91.06

Table 9. Quantitative comparison of UnionST Variants and the experimental setup is consistent with that of the main experiment.

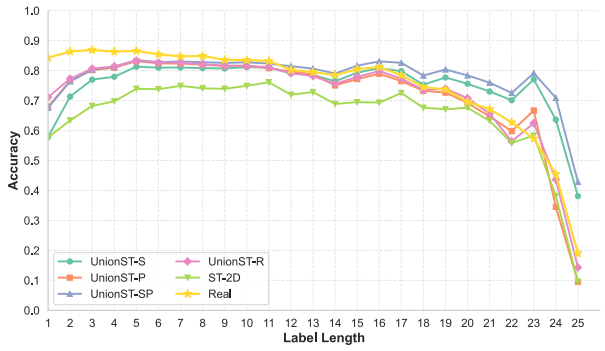

Figure 11. Accuracy as a function of label length on the General subset of the Union14M-Benchmark, evaluated using models trained on different datasets.

text, due to greater character confusion and error propagation. While models trained on real data perform well on short text, their effectiveness on longer labels is limited by the scarcity of such samples. In contrast, models trained on UnionST-S and UnionST-SP demonstrate greater robustness on longer labels, highlighting the effectiveness of our proposed data for complex and lengthy text recognition.

10.2. Comparisons with real data on STR models

Tab. 8 reports the performance gap between synthetic data (UnionST) and real datasets across various STR models. UnionST consistently outperforms real data on Multi-orientated and Multi-words subsets for most models. Notably, the SVTR [5] model achieves a higher average accuracy on UnionST than on real data. Although synthetic datasets still exhibit some differences compared to real data overall, UnionST significantly narrows this gap. Furthermore, UnionST, through its self-evolution semi-supervised framework, reduces reliance on labeled real data while achieving comparable performance.

10.3. Experiments on UnionST Variants

The results in Tab. 9 indicate that, at the same scale, UnionST-SP achieves an average accuracy only 0.12% lower than the ideal UnionST-SR, which fully leverages both large-scale synthetic and real corpora, on the Union14M-Benchmark. Notably, since pseudo-labeling introduces new corpus that closely match the real data distribution, combining these samples with the real dataset enables UnionST-SP to surpass UnionST-SR, yielding a 0.27% improvement in average accuracy. This further demonstrates the advantages of incorporating pseudo-labeling.

UnionST-PR represents the best performance that UnionST can achieve when the real dataset is known and used for generation. Even when the dataset size reaches 10M, the accuracy improvement slows down, revealing the scaling limit, and it still cannot outperform the real dataset. It is important to note that UnionST-PR serves as an idealized result, since our SEL framework can't access large-scale real labels. Under the setting where the real dataset is already available and used for pre-training, UnionST-SP maintains better performance than UnionST-PR. Rather than strictly matching the real distribution, UnionST-SP complements the real dataset, thereby achieving optimal performance.

10.4. Backbone-agnostic Gain

Using different backbones all leads to performance gains, we used SVTRv2-AR because it is a strong model, and the pursuit of optimal performance requires both high dataset quality and sufficient model capacity. In addition, we include results showing the gains brought by UnionST-S on other STR models (ABINet [11] and MAERec [19], see Tab. 10), which demonstrate that the improvements are backbone-agnostic.

Model	Training Data	Common Benchmarks							Union14M-Benchmark							
		IIIT	*SVT*	*IC13*	*IC15*	*SVTP*	*CUTE*	*AVG*	*CUR*	*MLO*	*ART*	*CTL*	*SAL*	*MLW*	*GEN*	*AVG*
ABINet [11]	U14M-Filter	98.50	98.10	97.70	90.10	94.10	96.50	95.83	80.40	69.00	71.70	74.70	77.60	76.80	79.80	75.72
	+ UnionST-S	99.10	98.30	98.83	90.89	94.42	97.22	96.46	86.31	92.40	75.78	80.62	82.25	85.92	81.04	83.48
	Δ	0.60	0.20	1.13	0.79	0.32	0.72	0.63	5.91	23.40	4.08	5.92	4.65	9.12	1.24	7.76
MAERec [19]	U14M-Filter	99.20	97.80	98.20	90.40	94.30	98.30	96.36	89.10	87.10	79.00	84.20	86.30	85.90	84.60	85.17
	+ UnionST-S	99.43	98.30	99.30	92.16	98.14	98.61	97.66	92.46	96.71	83.56	87.93	89.07	91.63	86.84	89.74
	Δ	0.23	0.50	1.10	1.76	3.84	0.31	1.30	3.36	9.61	4.56	3.73	2.77	5.73	2.24	4.57

Table 10. STR gains of UnionST-S by using different backbones.

10.5. Model Predictions Across Training Data

As shown in Fig. 12, predictions from models trained on UnionST are generally consistent with the ground truth. Notably, when the model trained on real data produces errors, UnionST-P often replicates these mistakes. However, UnionST-S frequently corrects such errors, underscoring the benefit of integrating both approaches in UnionST-SP. This combined strategy achieves performance comparable to models trained on real data, while mitigating certain biases inherent in real datasets. For instance, in the "NATONAL" example, the letter "I" is occluded. Both the real dataset and UnionST-P tend to hallucinate the missing character, whereas UnionST-S and UnionST-SP adhere strictly to the visual evidence and do not insert the absent letter.

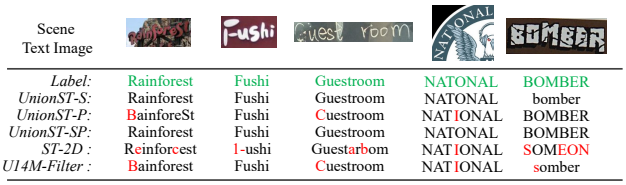

Figure 12. Visualization of model predictions on the Union14M-Benchmark, comparing models trained with different datasets. Characters differing from the ground truth are highlighted in red.

10.6. Hard Cases

Fig. 13 presents representative failure cases, which can be categorized as follows: (1) **Font-induced character similarity**: Artistic fonts can obscure distinctions between characters, as in the first example where "G" and "C" are easily confused, or in the fifth example where "L" resembles "P". (2) **Visual-semantic trade-off**: Some characters, such as "|" and "1", are visually ambiguous and require contextual understanding. Modeling such semantics about time remains challenging due to the scarcity of relevant training data. (3) **Ambiguity from occlusion or cropping**: In the third example, "P" is misrecognized as "F" due to partial occlusion, and the lack of context makes both interpretations plausible. (4) **Label noise**: The fourth example demonstrates a labeling error, where the ground truth should be "ASSGC" instead of "ASSGO", causing the correct prediction to be marked as incorrect. (5) **Extremely challenging cases**: In the sixth example, severe image blur renders the text nearly illegible, making accurate recognition difficult even for human annotators.

Figure 13. Examples of incorrect predictions by the best-performing model (average accuracy: 91.39% on Union14M-Benchmark).

11. Discussion

11.1. Emulation of Real-world

Even with synthetic backgrounds, we can create highly valuable training data that effectively emulates real-world variability and applicability, as demonstrated by the improved results in Tab. 4. For example, occlusions is modeled with the "Incomplete" operation, and multi-line text shares the same pipeline with vertical text (a special case of multi-line layout). We acknowledge that there are still differences compared to fully realistic scene images. However, purely real data is hard to balance and scale, and combining real and synthetic data becomes a clear trend to the OCR community. Then, the SEL further narrows the synthetic gap in real scenes while requiring only limited human effort. Moreover, UnionST can serve as a practical foundation for scalable, labeled full-scene synthesis, e.g., it first produces accurately labeled "sketches" with diverse layouts, then generative models refine these into photorealistic full scenes.

11.2. Multilingual Support

UnionST inherently supports multiple languages, and it can perform data synthesis as long as the corresponding language corpus is provided. As illustrated in Fig. 14, UnionST is capable of generating multilingual examples with significant diversity and realism. In our next steps,

we will develop a multilingual version of UnionST. We also plan to construct a similar multilingual evaluation benchmark following the "U14M-Bench" creation protocol, which covers diverse and challenging scenarios, to assess the utility of UnionST.

Arabic **Korean** **Japanese** **Chinese** **Hindi** **Russian**

Figure 14. Visualization of synthesized multilingual examples.

11.3. Consideration for Using SEL

Our goal is to verify the capability of the UnionST data engine. Therefore, we adopt SEL, a simple yet effective self-/semi-supervised learning baseline. Even in this straightforward setting, UnionST already works effectively and substantially reduces the annotation cost. UnionST can be seamlessly integrated with other self-/semi-supervised methods and we believe that incorporating more advanced methods would further improve the performance. Exploring such combinations is an important direction we plan to pursue.