SVTRv2: CTC Beats Encoder-Decoder Models in Scene Text Recognition

Supplementary Material

6. More Details of Ablation Study

SVTRv2 builds upon the foundation of SVTR by introducing several innovative strategies aimed at addressing challenges in recognizing irregular text and modeling linguistic context. The key advancements and their impact are detailed as follows:

Removal of the rectification Module and introduction of MSR and FRM. In the original SVTR, a rectification module is employed to recognize irregular text. However, this approach negatively impacts the recognition of long text. To overcome this limitation, SVTRv2 removes the rectification module entirely. To effectively handle irregular text without compromising the CTC model's ability to generalize to long text, MSR and FRM are introduced.

Improvement in feature resolution. SVTR extracts visual representations of size $\frac{H}{16} \times \frac{W}{4} \times D_2$ from input images of size $H \times W \times 3$. While this approach is effective for regular text, it struggles with retaining the distinct characteristics of irregular text. SVTRv2 doubles the height resolution ($\frac{H}{16} \rightarrow \frac{H}{8}$) of visual features, producing features of size $\frac{H}{8} \times \frac{W}{4} \times D_2$, thereby improving its capacity to recognize irregular text.

Refinement of local mixing mechanisms. SVTR employs a hierarchical vision transformer structure, leveraging two mixing strategies: Local Mixing is implemented through a sliding window-based local attention mechanism, and Global Mixing employs the standard global multi-head self-attention mechanism. SVTRv2 retains the hierarchical vision transformer structure and the global multi-head self-attention mechanism for Global Mixing. For Local Mixing, SVTRv2 introduces a pivotal change. Specifically, the sliding window-based local attention is replaced with two consecutive group convolutions (Conv^2) [21]. It is important to highlight that unlike previous CNNs, there is no normalization or activation layer between the two convolutions.

Semantic guidance module. The original SVTR model relies solely on the CTC framework for both training and inference. However, CTC is inherently limited in its ability to model linguistic context. SVTRv2 addresses this by introducing a Semantic Guidance Module (SGM) during training. SGM facilitates the visual encoder in capturing linguistic information, enriching the feature representation. Importantly, SGM is discarded during inference, ensuring that the efficiency of CTC-based decoding remains unaffected while still benefiting from its contributions during the training phase.

6.1. Progressive Ablation Experiments

To comprehensively evaluate the contributions of every SVTRv2 upgrade, a series of progressive ablation experiments are conducted. Tab. 7 outlines the results, along with the following observations:

1. Baseline (ID 0): The original SVTR serves as the baseline for comparison.

2. Rectification Module Removal (ID 1) reveals that while the rectification module (e.g., TPS) improves irregular text recognition accuracy, it hinders the model's ability to recognize long text. This confirms its limitations in balancing different recognition tasks.

3. Improvement in Feature Resolution (ID 2): Doubling the height resolution ($\frac{H}{16} \rightarrow \frac{H}{8}$) significantly boosts performance across challenging datasets, particularly for irregular text.

4. Replacement of Local Attention with Conv^2 (ID 3): Replacing the sliding window-based local attention with two consecutive group convolutions (Conv^2) yields improvements in artistic text, with a 3.0% increase in accuracy. This result highlights the efficacy of convolution-based approaches in capturing character-level nuances, such as strokes and textures, thereby improving its ability to recognize artistic and irregular text.

5. Incorporation of MSR and FRM (ID 4 and ID 5): These components collectively enhance accuracy on irregular text benchmarks (e.g., *Curve*), surpassing the rectification-based SVTR (ID 0) by 6.0%, without compromising the CTC model's ability to generalize to long text.

6. Integration of SGM (ID 6): Adding SGM yields significant gains on multiple datasets, improving accuracy on *OST* by 5.11% and *U14M* by 2.28%.

It can be summarized as that, by integrating Conv^2, MSR, FRM, and SGM, SVTRv2 significantly improves performance in recognizing irregular text and modeling linguistic context over SVTR, while still maintaining robust long-text recognition capabilities and preserving the efficiency of CTC-based inference.

7. SVTRv2 Variants

There are several hyper-parameters in SVTRv2, including the depth of channel (D_i) and the number of heads at each stage, the number of mixing blocks (N_i) and their permutation. By varying them, SVTRv2 architectures with different capacities could be obtained and we construct three typical ones, i.e., SVTRv2-T (Tiny), SVTRv2-S (Small), SVTRv2-B (Base). Their detail configurations are shown in Tab. 8.

In Tab. 8, $[L]_m[G]_n$ denotes that the first m mixing

ID	Method	*IIIT5k*	*SVT*	*ICDAR2013*	*ICDAR2015*	*SVTP*	*CUTE80*	Avg	*Curve*	*Multi-Oriented*	*Artistic*	*Contextless*	*Salient*	*Multi-Words*	*General*	Avg	*LTB*	*OST*	*Size*	*FPS*
		Common Benchmarks (*Com*)							Union14M-Benchmark (*U14M*)											
0	SVTR (w/ TPS)	98.1	96.1	96.4	89.2	92.1	95.8	94.62	82.2	86.1	69.7	75.1	81.6	73.8	80.7	78.44	0.0	71.2	19.95	141
1	0 + w/o TPS	98.0	97.1	97.3	88.6	90.7	95.8	94.58	76.2	44.5	67.8	78.7	75.2	77.9	77.8	71.17	45.1	67.8	18.10	161
2	$1 + \frac{H}{16} \rightarrow \frac{H}{8}$	98.9	97.4	97.9	89.7	91.8	96.9	95.41	82.2	64.3	70.2	80.0	80.9	80.6	80.5	76.95	44.8	69.5	18.10	145
3	2 + Conv^2	98.7	97.1	97.1	89.6	91.6	97.6	95.28	82.9	65.6	73.2	80.0	80.5	81.6	80.8	77.78	47.4	71.1	17.77	159
4	3 + MSR	98.7	98.0	97.4	89.4	91.6	97.6	95.44	87.4	83.7	75.4	80.9	81.9	83.5	82.8	82.22	50.9	72.5	17.77	159
5	4 + FRM	98.8	**98.1**	98.4	89.8	92.9	**99.0**	96.16	88.2	86.2	77.5	83.2	83.9	84.6	83.5	83.86	50.7	74.9	19.76	143
6	5 + SGM	**99.2**	98.0	**98.7**	**91.1**	**93.5**	**99.0**	**96.57**	**90.6**	**89.0**	**79.3**	**86.1**	**86.2**	**86.7**	**85.1**	**86.14**	**50.2**	**80.0**	19.76	143

Table 7. Ablation study of the proposed strategies on *Com* and *U14M*, along with their model sizes and FPS.

Models	$[D_0, D_1, D_2]$	$[N_1, N_2, N_3]$	Heads	Permutation
SVTRv2-T	[64,128,256]	[3,6,3]	[2,4,8]	$[L]_6[G]_6$
SVTRv2-S	[96,192,384]	[3,6,3]	[3,6,12]	$[L]_6[G]_6$
SVTRv2-B	[128,256,384]	[6,6,6]	[4,8,12]	$[L]_8[G]_{10}$

Table 8. Architecture specifications of SVTRv2 variants.

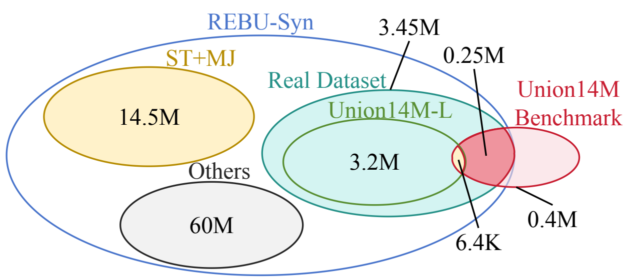

Figure 6. Relationships of the three real-world training sets and their overlapping with *U14M*.

blocks in SVTRv2 utilize local mixing, while the last *n* mixing blocks employ global mixing. Specifically, in SVTRv2-T and SVTRv2-S, all blocks in the first stage and the first three blocks in the second stage use local mixing. The last three blocks in the second stage, as well as all blocks in the third stage, are global mixing. In the case of SVTRv2-B, all blocks in the first stage and the first two blocks in the second stage use local mixing, whereas the last four blocks in the second stage and all blocks in the third stage adopt global mixing.

Algorithm 1: Inference Time

Input : A set of images $\mathcal{I}$ with size $|\mathcal{I}| = 3000$, batch size $B = 1$, N text lengths

Output: Overall inference time of the model

Initialize two lists: `total_time_list` and `count_list` of size N, initialized to 0;

for *each image* I_j *in* $\mathcal{I}$ *where* $j \in \{1, 2, \ldots, 3000\}$ **do**

  Determine the text length l_i for image I_j;

  Perform inference on I_j with text length l_i;

  Record inference time t_{ij};

  `total_time_list[`l_i`]` += t_{ij};

  `count_list[`l_i`]` += 1;

Initialize `avg_time_list`;

for *each text length* l_i *where* $i \in \{1, 2, \ldots, N\}$ **do**

  if *`count_list[`i`]`* > 0 **then**

    `avg_time_list[`i`]` = `total_time_list[`i`]` / `count_list[`i`]`;

Compute the final average inference time:

$$\texttt{inference\_time} = \frac{1}{N} \sum_{i=1}^{N} \texttt{avg\_time\_list}[i]$$

return `inference_time`;

8. More Details of Real-World Datasets

For English recognition, we train models on real-world datasets, from which the models exhibit stronger recognition capability [4, 25, 37]. There are three large-scale real-world training sets, i.e., the *Real* dataset [4], *REBU-Syn* [37], and *Union14M-L* (*U14M-Train*) [25]. However, as shown in Fig. 6 and Tab. 9, the former two significantly overlap with *U14M*, thus not suitable for model training when using *U14M* at the evaluation dataset. Surprisingly, *U14M-Train* is also overlapped with *U14M* in nearly 6.5k text instances across the seven subsets. It means the models trained based on *U14M-Train* suffer from data leakage when tested on *U14M*, thus the results reported by [25] should be updated. To this end, we create a filtered version of *Union14M-L*, termed as *U14M-Filter*, by filtering these overlapping instances from the training set. This new dataset is used to train SVTRv2 and other 24 methods we reproduced.

	Curve 2,426	*Multi-Oriented* 1,369	*Artistic* 900	*Contextless* 779	*Salient* 1,585	*Multi-Words* 829	*General* 400,000
Real [4]	1,276	440	432	326	431	193	254,174
REBU-Syn [37]	1,285	443	462	363	442	289	260,575
U14M-Train [25]	9	3	30	37	11	96	6,401

Table 9. Overlapping statistics between three real-world training sets and *U14M*.

9. More Details of Inference Time

In terms of the inference time, we do not utilize any acceleration framework and instead employ PyTorch's dynamic graph mode on one NVIDIA 1080Ti GPU. We first measure the inference time for 3,000 images with a batch size of *1*, calculating the average inference time for each text length. We then compute the arithmetic mean of the average time across all text lengths to determine the overall inference time of the model. Algorithm 1 details the process of measuring inference time.

10. Results when Trained on Synthetic Datasets

Previous research typically follows a typical evaluation protocol, where models are trained on synthetic datasets and validated using *Com*, the six widely recognized real-world benchmarks. Following this protocol, we also train SVTRv2 and other models on synthetic datasets. In addition to evaluating SVTRv2 on *Com*, we assess its performance on *U14M*. The results offer a comprehensive evaluation of the model's generalization capabilities. For methods that have not reported performance on challenging benchmarks, we conduct additional evaluations using their publicly available models and present these results for comparative analysis. As illustrated in Tab. 10, models trained on synthetic datasets exhibit notably lower performance compared to those trained on large-scale real-world datasets (see Tab. 3). This performance drop is particularly pronounced on challenging benchmarks. These findings highlight the critical importance of real-world datasets in improving recognition accuracy.

Despite trained on less diverse synthetic datasets, SVTRv2 also exhibits competitive performance. On irregular text benchmarks, such as *Curve* and *Multi-Oriented*, SVTR achieves strong results, largely due to its integrated rectification module [40], which is particularly adept at handling irregular text patterns, even when trained on synthetic datasets. Notably, SVTRv2 achieves a substantial 4.8% improvement over SVTR on *Curve*, further demonstrating its enhanced capacity to address irregular text. Overall, these results demonstrate that, even when trained solely on synthetic datasets, SVTRv2 exhibits strong generalization capabilities, effectively handling complex and challenging text recognition scenarios.

11. Qualitative Analysis of Recognition Results

The SVTRv2 model achieved an average accuracy of 96.57% on *Com* (see Tab. 3). To investigate the underlying causes of the remaining 3.43% of recognition errors, we conducted a detailed analysis of the misclassified samples, as illustrated in Fig. 7 and Fig. 8. While previous research has typically categorized *Com* into *regular* and *irregular* text. However, these error samples indicate that the majority of incorrectly recognized text is not irregular. This suggests that, under the current training paradigm using large-scale real-world datasets, a more rigorous manual screening process is warranted for common benchmarks.

Based on this one-by-one manual viewing, we identified five primary causes of recognition errors: (1) blurred, (2) artistic, (3) incomplete text, (4) others, and (5) image text labeling errors (Label_{err}). Specifically, the blurring text includes issues such as low resolution, motion blur, or extreme lighting conditions. The artistic text category refers to unconventional fonts, commonly found in business signage, as well as some handwritten text. Incomplete text arises when characters are obscured by objects or lost due to improper cropping, requiring contextual inference. Image text labeling errors occur when the given text labels contain inaccuracies or include characters with phonetic symbols. As shown in Tab. 11, after excluding samples affected by labeling inconsistencies, the remaining recognition errors primarily stemmed from blurred (30.81%), artistic (24.24%), and incomplete text (31.82%). This result highlights that SVTRv2's recognition performance needs further improvement, particularly in handling complex scenarios involving these challenging text types.

12. Standardized Model Training Settings

The optimal hyperparameters for training different models vary and are not universally fixed. However, key factors such as training epochs, data augmentations, input size, and evaluation protocols significantly influence model accuracy. To ensure fair and unbiased performance comparisons, we standardize these factors across all models, as outlined in Tab. 12. This uniform training and evaluation framework ensures consistency while allowing each model to approach its best accuracy. To maximize fairness, we conducted extensive hyperparameter tuning for model-specific settings, including the optimizer, learning rate, and regularization

Method	Venue	Encoder	Common Benchmarks (*Com*)						Avg	Union14M-Benchmark (*U14M*)							Avg	*Size*
			IIIT5k	*SVT*	*ICDAR2013*	*ICDAR2015*	*SVTP*	*CUTE80*		*Curve*	*Multi-Oriented*	*Artistic*	*Contextless*	*Salient*	*Multi-Words*	*General*		
ASTER [40]	TPAMI 2019	ResNet+LSTM	93.3	90.0	90.8	74.7	80.2	80.9	84.98	34.0	10.2	27.7	33.0	48.2	27.6	39.8	31.50	27.2
NRTR [38]	ICDAR 2019	Stem+TF$_6$	90.1	91.5	95.8	79.4	86.6	80.9	87.38	31.7	4.40	36.6	37.3	30.6	54.9	48.0	34.79	31.7
MORAN [32]	PR 2019	ResNet+LSTM	91.0	83.9	91.3	68.4	73.3	75.7	80.60	8.90	0.70	29.4	20.7	17.9	23.8	35.2	19.51	17.4
SAR [29]	AAAI 2019	ResNet+LSTM	91.5	84.5	91.0	69.2	76.4	83.5	82.68	44.3	7.70	42.6	44.2	44.0	51.2	50.5	40.64	57.7
DAN [46]	AAAI 2020	ResNet+FPN	93.4	87.5	92.1	71.6	78.0	81.3	83.98	26.7	1.50	35.0	40.3	36.5	42.2	42.1	32.04	27.7
SRN [55]	CVPR 2020	ResNet+FPN	94.8	91.5	95.5	82.7	85.1	87.8	89.57	63.4	25.3	34.1	28.7	56.5	26.7	46.3	40.14	54.7
SEED* [36]	CVPR 2020	ResNet+LSTM	93.8	89.6	92.8	80.0	81.4	83.6	86.87	40.4	15.5	32.1	32.5	54.8	35.6	39.0	35.70	24.0
AutoSTR* [59]	ECCV 2020	NAS+LSTM	94.7	90.9	94.2	81.8	81.7	-	-	47.7	17.9	30.8	36.2	64.2	38.7	41.3	39.54	6.00
RoScanner [57]	ECCV 2020	ResNet	95.3	88.1	94.8	77.1	79.5	90.3	87.52	43.6	7.90	41.2	42.6	44.9	46.9	39.5	38.09	48.0
ABINet [15]	CVPR 2021	ResNet+TF$_3$	96.2	93.5	97.4	86.0	89.3	89.2	91.93	59.5	12.7	43.3	38.3	62.0	50.8	55.6	46.03	36.7
VisionLAN [47]	ICCV 2021	ResNet+TF$_3$	95.8	91.7	95.7	83.7	86.0	88.5	90.23	57.7	14.2	47.8	48.0	64.0	47.9	52.1	47.39	32.8
PARSeq* [4]	ECCV 2022	ViT-S	97.0	93.6	97.0	86.5	88.9	92.2	92.53	63.9	16.7	52.5	54.3	68.2	55.9	56.9	52.62	23.8
MATRN [34]	ECCV 2022	ResNet+TF$_3$	96.6	95.0	97.9	86.6	90.6	93.5	93.37	63.1	13.4	43.8	41.9	66.4	53.2	57.0	48.40	44.2
MGP-STR* [45]	ECCV 2022	ViT-B	96.4	94.7	97.3	87.2	91.0	90.3	92.82	55.2	14.0	52.8	48.5	65.2	48.8	59.1	49.09	148
LevOCR* [9]	ECCV 2022	ResNet+TF$_3$	96.6	94.4	96.7	86.5	88.8	90.6	92.27	52.8	10.7	44.8	51.9	61.3	54.0	58.1	47.66	109
CornerTF* [51]	ECCV 2022	CornerEncoder	95.9	94.6	97.8	86.5	91.5	92.0	93.05	62.9	18.6	56.1	58.5	68.6	59.7	61.0	55.07	86.0
SIGA* [18]	CVPR 2023	ViT-B	96.6	95.1	97.8	86.6	90.5	93.1	93.28	59.9	22.3	49.0	50.8	66.4	58.4	56.2	51.85	113
CCD* [19]	ICCV 2023	ViT-B	97.2	94.4	97.0	87.6	91.8	93.3	93.55	66.6	24.2	**63.9**	64.8	74.8	62.4	64.0	60.10	52.0
LISTER* [8]	ICCV 2023	FocalNet-B	96.9	93.8	97.9	87.5	89.6	90.6	92.72	56.5	17.2	52.8	63.5	63.2	59.6	65.4	54.05	49.9
LPV-B* [58]	IJCAI 2023	SVTR-B	97.3	94.6	97.6	87.5	90.9	94.8	93.78	68.3	21.0	59.6	65.1	76.2	63.6	62.0	59.40	35.1
CDistNet* [65]	IJCV 2024	ResNet+TF$_3$	96.4	93.5	97.4	86.0	88.7	93.4	92.57	69.3	24.4	49.8	55.6	72.8	64.3	58.5	56.38	65.5
CAM* [54]	PR 2024	ConvNeXtV2-B	97.4	**96.1**	97.2	87.8	90.6	92.4	93.58	63.1	19.4	55.4	58.5	72.7	51.4	57.4	53.99	135
BUSNet [49]	AAAI 2024	ViT-S	96.2	95.5	98.3	87.2	91.8	91.3	93.38	-	-	-	-	-	-	-	-	56.8
DCTC [60]	AAAI 2024	SVTR-L	96.9	93.7	97.4	87.3	88.5	92.3	92.68	-	-	-	-	-	-	-	-	40.8
OTE [52]	CVPR 2024	SVTR-B	96.4	95.5	97.4	87.2	89.6	92.4	93.08	-	-	-	-	-	-	-	-	25.2
CPPD [13]	TPAMI 2025	SVTR-B	97.6	95.5	98.2	87.9	90.9	92.7	93.80	65.5	18.6	56.0	61.9	71.0	57.5	65.8	56.63	26.8
IGTR-AR [14]	TPAMI 2025	SVTR-B	**98.2**	95.7	**98.6**	**88.4**	**92.4**	95.5	**94.78**	**78.4**	31.9	61.3	66.5	**80.2**	69.3	**67.9**	**65.07**	24.1
SMTR [12]	AAAI 2025	FocalSVTR	97.4	94.9	97.4	**88.4**	89.9	**96.2**	94.02	74.2	30.6	58.5	67.6	79.6	**75.1**	**67.9**	64.79	15.8
CRNN [39]	TPAMI2016	ResNet+LSTM	82.9	81.6	91.1	69.4	70.0	65.5	76.75	7.50	0.90	20.7	25.6	13.9	25.6	32.0	18.03	8.30
SVTR* [11]	IJCAI2022	SVTR-B	96.0	91.5	97.1	85.2	89.9	91.7	91.90	69.8	**37.7**	47.9	61.4	66.8	44.8	61.0	55.63	24.6
SVTRv2	-	SVTRv2-B	97.7	94.0	97.3	88.1	91.2	95.8	94.02	74.6	25.2	57.6	**69.7**	77.9	68.0	66.9	62.83	19.8

Table 10. Results of SVTRv2 and existing models when trained on synthetic datasets (*ST* + *MJ*) [20, 24]. * represents that the results on *U14M* are evaluated using the model they released.

	Blurred	Artistic	Incomplete	Other	Total	Label$_{err}$
IIIT5k [33]	0	16	1	4	21	4
SVT [44]	4	4	4	0	12	0
ICDAR 2013 [27]	2	2	4	2	10	2
ICDAR 2015 [26]	48	19	42	13	122	35
SVTP [35]	7	6	12	7	32	4
CUTE80 [1]	0	1	0	0	1	1
Total	61	48	63	26	198	46
	30.81%	24.24%	31.82%	13.13%	100%	

Table 11. Distribution of bad cases for SVTRv2 on *Com*.

strategies. This rigorous optimization led to significant accuracy improvements of 5–10% for most models compared to their default configurations. For instance, MAERec's accuracy increased from 78.6% to 85.2%, demonstrating the effectiveness of training settings. These improvements underscore the reliability of our results and highlight the importance of carefully optimizing hyperparameters for meaningful model comparisons.

Setting	Detail
Training Set	For training, when the text length of a text image exceeds 25, samples with text length ≤ 25 are randomly selected from the training set to ensure models are only exposed to short texts (length ≤ 25).
Test Sets	For all test sets except the long-text test set (*LTB*), text images with text length > 25 are filtered. Text length is calculated by removing spaces and non-94-character-set special characters.
Input Size	Unless a method explicitly requires a dynamic size, models use a fixed input size of 32×128. If a model performs incorrectly with 32×128 during training, the original size is used. The test input size matches the training size.
Data Augmentation	All models use the data augmentation strategy employed by PARSeq.
Training Epochs	Unless pre-training is required, all models are trained for 20 epochs.
Optimizer	AdamW is the default optimizer. If training fails to converge with AdamW, Adam or other optimizers are used.
Batch Size	Maximum batch size for all models is 1024. If single-GPU training is not feasible, 2 GPUs (512 per GPU) or 4 GPUs (256 per GPU) are used. If 4-GPU training runs out of memory, the batch size is halved, and the learning rate is adjusted accordingly.
Learning Rate	Default learning rate for batch size 1024 is 0.00065. The learning rate is adjusted multiple times to achieve the best results.
Learning Rate Scheduler	A linear warm-up for 1.5 epochs is followed by a OneCycle scheduler.
Weight Decay	Default weight decay is 0.05. NormLayer and Bias parameters have a weight decay of 0.
EMA or Similar Tricks	No EMA or similar tricks are used for any model.
Evaluation Protocols	Word accuracy is evaluated after filtering special characters and converting all text to lowercase.

Table 12. A uniform training and evaluation setting to maintain consistency across all settings while simultaneously enabling each model to achieve its best possible accuracy.

Figure 7. The bad cases of SVTRv2 in IIIT5k [33], SVT [44], ICDAR 2013 [27], SVTP [35] and CUTE80 [1]. Labels, the predicted result, and the predicted score are denoted as Text_{label} | Text_{pred} | Score_{pred}. Yellow, red, blue, and green boxes indicate blurred, artistic fonts, incomplete text, and label-inconsistent samples, respectively. Other samples have no box.

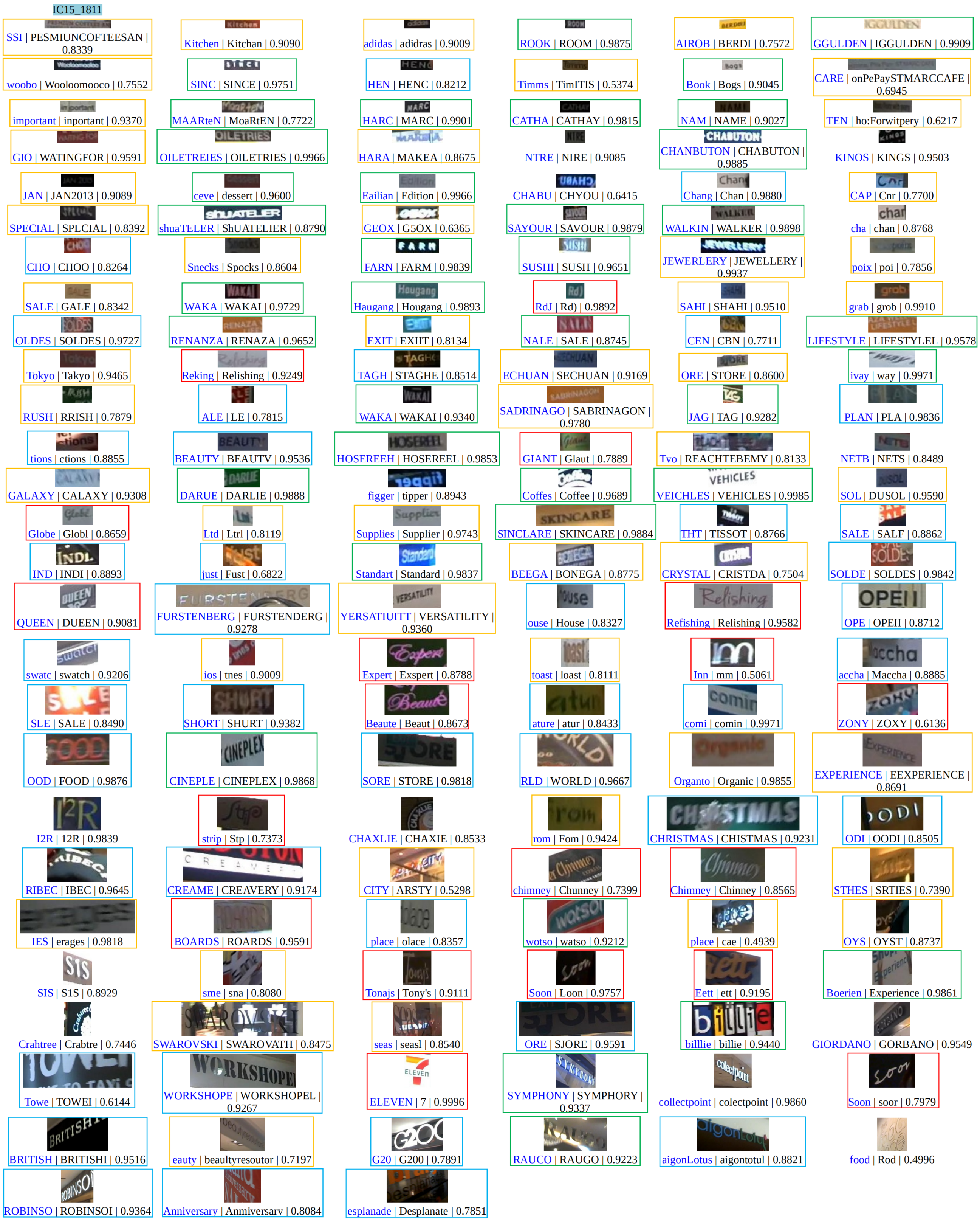

Figure 8. The bad cases of SVTRv2 in ICDAR 2015 [26]. Labels, the predicted result, and the predicted score are denoted as Text_{label} | Text_{pred} | Score_{pred}. Yellow, red, blue, and green boxes indicate blurred, artistic fonts, incomplete text, and label-inconsistent samples, respectively. Other samples have no box.