Supplementary Materials for CFOR: Character-First Open-Set Text Recognition via Context-Free Learning

Chang Liu [a,b], Chun Yang*[a], Zhiyu Fang[a], Hai-Bo Qin[a], Xu-Cheng Yin[a], *Senior Member, IEEE*

I. Data Specifics

This work involves a lot of data sources, which we detail here together with a brief description.

A. Close Set Experiments

For close-set text recognition experiments, we train the model with the standard [1] MJ [2] and ST [3] synthetic datasets. In this work, we use the datasets from DAN [4]. We choose the following dataset to benchmark the model for close-set performance.

a) IIIT5k [5]: has 3000 images for testing, including English word clips from several sources, mainly scene and born-digital images.

b) CUTE80 [6]: has 80 images for testing, including curved scene text samples. Despite its small size, it is a popular dataset to measure OCR performance on curvy text [1], [7], [8].

c) SVT [9]: has 647 images collected from street view images, mainly on informative signs like road signs, shop names, etc.

d) ICDAR2003(IC03) [10]: has 867 images for testing, mostly well-focused and aligned scene text crops.

e) ICDAR2013(IC13) [11]: has 1015 images for testing, and is a superset of IC03.

B. Open-set Experiments

Following Liu et al. [12], we use the horizontal clips that only contains Tire-1 Chinese characters, English Characters, and digits from the listed datasets to train our model:

a) ART [13]: is a dataset including arbitrary shaped words from $10,166$ images.

b) RCTW [14]: is a dataset with $12,263$ mixed natural-scene and born-digital images.

c) LSVT [15]: is a huge street view dataset with different levels of annotations. Here we only take the $50,000$ images with full annotation.

d) MLT [16]: is a multilingual scene text dataset including 9 languages, each has $20,000$ images. Here, we pick the Latin part and the Chinese part to train our model.

[a]School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, China.
[b] ML-Group, Luleå Tekniska Universitet, Sweden.
* Corresponding Author

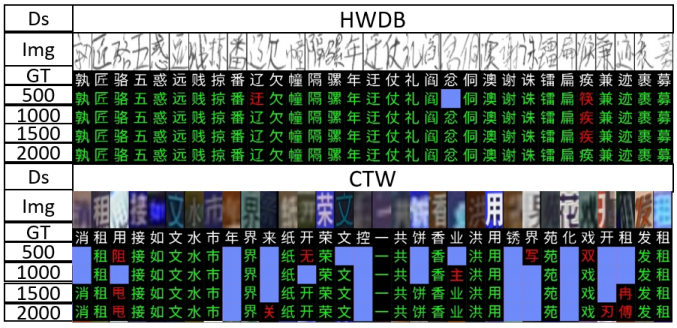

Fig. 1. Sample results on HWDB and CTW datasets. Number on the left indicates the number of training samples, white indicates the ground truth, green indicates correct prediction, and purple block indicates rejected as unknown unknown samples.

e) CTW [17]: As previously introduced, the CTW dataset is a multi-level annotated dataset

For quantitative testing, we use the Japanese subset and the Korean subset to quantitatively test the recognition and out-of-set spotting capability. For qualitative results, we use the data from SIW-13 [18] and images cropped from Google search. SIW13 is a language identification dataset without content annotation.

C. Zero-shot Character Recognition Experiments

For zero-shot Chinese character recognition, we adopt the protocol from HDE [19] following Liu's work [12], which involves two datasets,

a) HWDB [20]: includes $3.9M$ handwritten characters. The dataset includes 3 training subsets, which are HWDB-1.0, HWDB-1.1, and HWDB-1.2. The dataset also includes a testing subset used for the ICDAR 2013 competition, including $224K$ samples collected from 60 writers.

b) CTW [17]: includes $1M$ characters taken from scene images, where some inevitably suffer from different levels and types of degeneration including blur, skew, occlusion, etc.

II. Extra Experiments

A. Zero-shot Character Recognition

Zero-shot character recognition is also a notable special case of the open-set text recognition task, where $\mathbf{C}^u_{test} = \varnothing$, $\mathbf{C}^k_{test} \cap \mathbf{C}_{train} = \varnothing$, and the length of all words equal to one. To provide a referenced comparison of generalization capability on NICs, we also benchmark our framework following

TABLE I
ZERO-SHOT CHARACTER RECOGNITION ACCURACY ON HWDB AND CTW DATASETS. * INDICATES "ONLINE TRAJECTORY" DATA REQUIRED.

Method	Venue	Accuracy (%)							
		HWDB				CTW			
		# characters in training set				# characters in training set			
		500	1000	1500	2000	500	1000	1500	2000
CM* [21]	ICDAR'19	44.68	71.01	80.49	86.73	-	-	-	-
FewRan [22]	PRL'19	33.6	41.5	63.8	70.6	2.36	10.49	16.59	22.03
HCCR [19]	PR'20	33.71	53.91	66.27	73.42	23.53	38.47	44.17	49.79
OSOCR [12]	PR'23	47.92	74.02	81.11	85.72	28.03	49.00	58.37	64.03
Ours	-	**89.78**	**93.42**	**94.70**	**95.07**	**55.63**	**60.19**	**68.31**	**73.03**

the convention of the community [19], [22]. Specifically, we run benchmarks on the HWDB and the CTW datasets, and split the training character set $\mathbf{C}_{train}$ and evaluation set $\mathbf{C}_{test}$ following HCCR [19] and OSOCR [12]. [1] Specifically, for each dataset, we first sample the testing labels (500 for CTW and 1000 for HWDB), then sample 500 to 2000 labels from the **remaining** characters for training.[2] The models are trained for $80k$ iterations for each setup on both datasets, and characters in the evaluation sets are excluded from the Individual Character Learning task to preserve the inductive setup. The qualitative results are shown in Fig. 1, and quantitative results are shown in Table I. Qualitative results on HWDB demonstrate reasonable robustness against different writing styles, while on the CTW dataset the model shows overall robustness on slight blur and affine transformation. The robustness improves alongside the expansion of the training label set. The model also tends to reject very hard samples with heavy blur and occlusion instead of recognizing them as completely different characters. The overall robustness can also be validated by the quantitative results. Compared to existing zero-shot character recognition methods, our framework demonstrates strong generalization capability on all setups of both datasets. The results also indicate the framework work reasonably well when contextual information does not apply to the training data.

An generalized open-set recognition [24] variant of the task is also performed to validate the rejection capability as per OSOCR [12]. The models trained with 2000 characters in the previous experiments are benchmarked with different portions of $\mathbf{C}_{test}$ split into $\mathbf{C}^u_{test}$. Quantitative results that measure the effectiveness of spotting unseen characters are shown in Table II. Our framework demonstrates overall robustness against size changes of the character set $|\mathbf{C}^k_{test}|$, i.e. the Accuracy (Acc.) does not vary much as $\mathbf{C}^k_{test}$ grows. The Recall, Precision, and F-measure drop slightly as $|\mathbf{C}^u_{test}|$ grows, however, they remain on decent levels. The drop in Precision is caused by the rejection of blur samples, which is expected, as precision would be zero when $|\mathbf{C}^u_{test}| = \varnothing$. However, the dropping recall (Rec.) indicates an increasing tendency for an out-of-set sample to mistakenly fall into some in-set classes' decision boundaries. The plural is used here as it is possible for the sample falls in the intersection part of more than one class. This suggests that the would-be decision boundaries of different unknown classes may still overlap, which remains to be solved in the future.

[1]Note this split is not directly comparable to another popular protocol [23] which trains on "simple" characters and tests on "hard" ones, which caused the significant gap of HCCR between the original paper [19] and the performance reported by [23].

[2]The training label set $\mathbf{C}_{train}$ is disjoint to the testing label set $\mathbf{C}^k_{test}$, and the detailed splits are released with our code.

TABLE II
REJECTION PERFORMANCE ON HWDB AND CTW DATASETS.

	HWDB				CTW			
#NIC	100	200	400	500	50	100	200	250
#NOC	900	800	600	500	450	400	300	250
Rec.	95.9	91.5	85.3	82.3	98.5	96.1	90.9	88.9
Pre.	99.5	99.2	98.0	97.2	98.2	92.8	85.9	80.1
F	97.7	95.2	91.2	89.1	98.3	94.4	88.3	84.3
Acc.	96.0	96.7	96.0	96.0	74.4	76.7	75.2	73.4

B. Different Designs of Contextual Information Removal

This work focuses on alleviating the confounding effect of contextual information by isolating characters from their context. Alternatively, one can co-train with randomly generated synthetic data, which is not confounded by the context, to achieve the same goal. We also briefly validated the idea and the results are shown in Table III. An ICL-only model is trained along with the exact same real data stream to offer a controlled comparison. Performance-wise, using randomly generated strings does not yield noticeable improvement against using individual characters. However, it yields more computation costs during training, hence we consider the ICL module slightly more feasible compared to using synthetic word-level data from random strings. In addition, the second run of the CIL-only model trained on a 3090 demonstrates a marginal performance difference from the first run trained on 1080Ti in the ablative machine, showing a reasonable reproduction capability.

C. The Necessity of Domain Specific Batch Norm

In this work, the ICL task is designed as a stand-alone task using its own batch statistic. The design is necessary to avoid the adverse effect that we call "performance bleeding" (Fig. 2), caused by the domain bias of single synthetic characters and word-level training data. To demonstrate this effect, we conducted experiments with different sharing schemes, namely "share-1" and "share-2". The "share-1" scheme simulates mixing word-level and character-level data during training, by setting BN-1=BN-4 and BN-2=BN-5 in Fig. **??**. "Share-2" simply shares batch statistics among all auxiliary tasks, by setting BN-3=BN-4=BN-5. The results show that both schemes

TABLE III
COMPARING CO-TRAINING WITH INDIVIDUAL CHARACTER AND SYNTHETIC RANDOM STRINGS.

Split	Name	ICL	LA	Recall	Precision	F-measure
GZSL	Individual Character Learning-run2	single character	**41.18**	-	-	-
	Random Word Learning	random string	40.48	-	-	-
OSR	Individual Character Learning-run2	single character	72.07	**64.29**	96.16	**77.06**
	Random Word Learning	random string	**73.83**	63.93	**96.19**	76.81
GOSR	Individual Character Learning-run2	single character	**65.64**	**52.35**	83.15	64.25
	Random Word Learning	random string	65.60	51.62	**85.10**	**64.26**
OSTR	Individual Character Learning-run2	single character	**69.87**	77.68	90.24	83.49
	Random Word Learning	random string	68.72	**79.20**	**92.58**	**85.37**

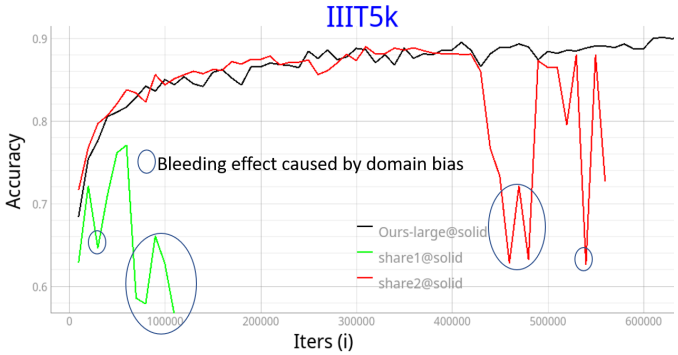

Fig. 2. The bleeding effect caused by the domain bias between word-level data and character level data when batchnorm layers are shared among different domains. Share 1: ICL shares BN with backbone (BN-1=BN-4, BN-2=BN-5), Share 2: ICL shares BN with CIL (BN-3=BN-4=BN-5).

suffer from serious performance degradation. Alike effects are also studied in the multi-task learning community [25]. Considering the domain gap between synthetic character data and real-word data, simply mixing data and training the model is not a feasible design. Hence in this paper, we implement ICL as an auxiliary training task with its own batch statics.

III. DISCUSSIONS

A. Limitations

Despite showing reasonable robustness on modern scripts from the CJK family and the Greek family, the model starts showing strong limitations on the Glagolitic samples and shows no generalization capability toward the scripts like Hindi (Devanagari) and Bangladesh (Bengali). In essence, we are trying to fit two functions, mapping either samples or labels representations (glyphs) into a common feature space. However, the training set is limited on both class numbers and word samples, thus both can and will overfit, consequentially limits the generalization capability.

Specifically, discriminative visual features from the training set are not complete. Discriminative representations, esp. detailed ones, may not be modeled when they do not play a significant role during training. For example, rectangular and circle can be considered as a mere difference in style for English and Chinese, however for Korean they are two different components. Failure to model such kind differences leads to a severe performance gap between Unique Kanji, Korean, and Kanas albeit all of them are novel to the model. The ICL module can somehow improve the generalization

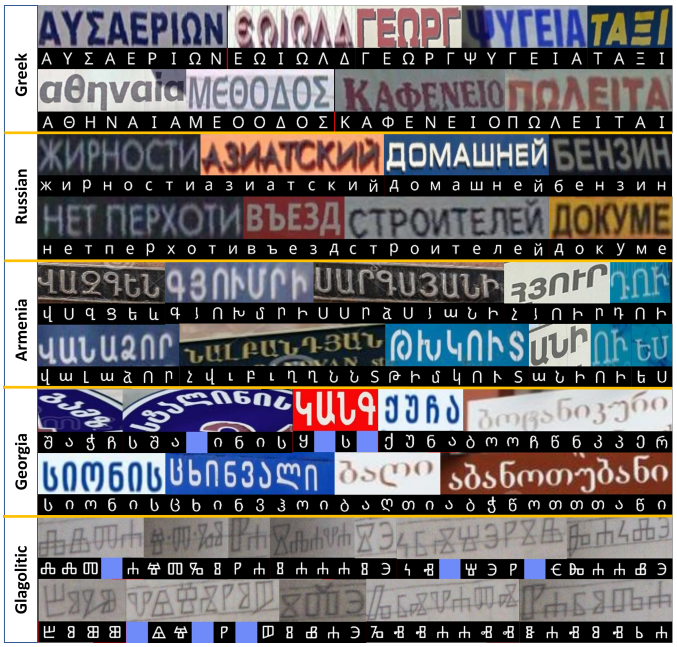

Fig. 3. Sample results from unannotated scripts of Greek family scripts collected from the SIW-13 dataset and the Internet. Each sample is composed with the orignial image and the OCR result (blue blocks indicate rejections).

capability (improved 3% on GZSL), however, the performance is still way lower than many-shot methods, indicating the overfitting still persists, and to be solved in the future.

Second, in the Glagolitic case in Fig. 3, we notice the attention module failed to locate characters accurately, resulting in repetitive predicted characters, likely caused by the L-CAM module overfits to the styles seen in the training set and failure to generalize to the rare stone carving style.

Finally, the performance is yet to be satisfactory compared to heavy SOTAs like PERN2D [26] and JVSR [27] due to the lack of language models, the small backbone, and the absence of rectification mechanisms [28], [29]. For the language model, an extra semantic-based stage [8], [30] can be simply added if the users are confident about the "testing set corpus" being well covered by the training set.

B. Difference between CIM and [31]

A similar complementary property is also used in [31], which aims to utilize context information via reconstructing the masked-off information like [32], [33]. On the contrary, the proposed Context Isolation mechanism aims to eliminate the contextual dependency in training data, which leads to

significant design differences in input handling, label handling, and attention handling, and (setting the masked character to unknown instead of leaving the labels untouched [31], [32]).

Input-wise, unlike [31] applying the mask on the feature map, the mask in CIM is directly applied to the original image, to prevent the information of the wiped contextual regions from being preserved as a result of the large reception field of the feature extractor.

Label-handling-wise, CIM setting the masked character to unknown instead of skipping them [31] or leaving the labels untouched [32].

Attention-handling-wise, the recognizer from CIM does not generate its own attention map, so that the masked regions are not skipped. The word-level sampler $\texttt{Att}_\texttt{t}$ needs to sample from where the masked characters were, so that the classifier can confirm that they are properly masked so that they produce the "unknown" label, instead of just computation a new set of attention maps and skip these characters like [31]

TABLE IV
NOTATION TABLE

Notation	type	description
$\hat{\boldsymbol{Y}}$	Sequence of confidence vectors	The predicted probablity sequence, each element corresponds to the probability of a character
$\boldsymbol{Y}$	Sequence of character	A sequence of character, could be ground truth or predicted results
A^{neg}	confidence map	The remaining regions not covered by A^{pos}. Could be either background or characters
$\tilde{S}$	tensor	Similarity of character feature at each timestamp t and each **prototype**.
M^c	tensor	An (intermediate) feature map of the input glyph.
S^c	tensor	Similarity of character feature at each timestamp t and each **character**.
$\mathbf{C}_{test}^k$	set	Labels in the testing set with side-information.
$\hat{\mathbf{Y}}_{pos}$	Set	The characters that are correctly predicted for a training sample.
$\mathbf{C}_{test}$	set	Set of distinct character(label) in the testing set
$\mathbf{C}_{train}$	set	Set of distinct character(label) in the training set
C	-	contextual information
$\mathbf{C}_{test}^u$	set	Out-of-set labels in the testing set without side-information.
A	tensor	.
M	tensor	An (intermediate) feature map of the input word clip.
F		The character representation of each character in one sample. Could be a feature, but could also be "attributes" as well.
A^{pos}	confidence map	Regions selected to mask off to break context.
$\mathtt{NR_p}$	Function	Ideal recognizer, which yields oracles.
img	tensor	An (intermediate) feature map of the input word clip.
G	-	side information
P	tensor	Prototypes for all classes. Note one class can be mapped to more than one prototypes depending on the number of cases.
ϕ	Function	Returns label assosiated to a prototype
t	index	indicates a timestamp (the t^{th} item)
i	Scalar	Indexing value
j	Scalar	Indexing value
$[-]$	special token	Special token indicating the unknown label
$\hat{\mathbf{Y}}_{neg}$	Set	The characters that are NOT correctly predicted for a training sample.

IV. NOTATION TABLE

There is a lot of symbols in the paper, so we have summarized them in Table. IV. We also list some of the styling guidelines:

$\mathtt{A}$ indicates $\mathtt{A}$ is a function.

$\mathbb{R}$ indicates $\mathbb{R}$ is a domain, e.g. real numbers, integers, etc.

$\mathcal{A}$ indicates $\mathcal{A}$ is a space, which means the set include all elements that meet a certain creteria.

$\mathbf{A}$ indicates $\mathbf{A}$ is a set.

$\boldsymbol{A}$ indicates $\boldsymbol{A}$ is an array.

a indicates a is an element, e.g. real number, character, or other more or less atomic elements.

$\hat{A}$ indicates $\hat{A}$ is explicitly prediction made by the model.

A^* indicates A^* is explicitly oracles from domain experts.

REFERENCES

[1] J. Baek, G. Kim, J. Lee *et al.*, "What is wrong with scene text recognition model comparisons? dataset and model analysis," in *ICCV*. IEEE, 2019, pp. 4714–4722.

[2] M. Jaderberg, K. Simonyan, A. Vedaldi *et al.*, "Synthetic data and artificial neural networks for natural scene text recognition," in *NeuIPS Workshop*. Neural Information Processing Systems, 2014.

[3] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *CVPR*. IEEE Computer Society, 2016, pp. 2315–2324.

[4] T. Wang, Y. Zhu, L. Jin *et al.*, "Decoupled attention network for text recognition," in *AAAI*. AAAI Press, 2020, pp. 12 216–12 224.

[5] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *BMVC*. BMVA Press, 2012, pp. 1–11.

[6] A. Risnumawan, P. Shivakumara, C. S. Chan *et al.*, "A robust arbitrary text detection system for natural scene images," *Expert Syst. Appl.*, vol. 41, no. 18, pp. 8027–8048, 2014.

[7] B. Shi, M. Yang, X. Wang *et al.*, "ASTER: an attentional scene text recognizer with flexible rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2035–2048, 2019.

[8] S. Fang, Z. Mao, H. Xie *et al.*, "Abinet++: Autonomous, bidirectional and iterative language modeling for scene text spotting." IEEE, 2022.

[9] K. Wang, B. Babenko, and S. J. Belongie, "End-to-end scene text recognition," in *ICCV*. IEEE Computer Society, 2011, pp. 1457–1464.

[10] S. M. Lucas, A. Panaretos, L. Sosa *et al.*, "ICDAR 2003 robust reading competitions: entries, results, and future directions," *Int. J. Document Anal. Recognit.*, vol. 7, no. 2-3, pp. 105–122, 2005.

[11] D. Karatzas, F. Shafait, S. Uchida *et al.*, "ICDAR 2013 robust reading competition," in *ICDAR*. IEEE Computer Society, 2013, pp. 1484–1493.

[12] C. Liu, C. Yang, H. Qin *et al.*, "Towards open-set text recognition via label-to-prototype learning," *Pattern Recognit.*, vol. 134, p. 109109, 2023.

[13] C. K. Chng, E. Ding, J. Liu *et al.*, "ICDAR2019 robust reading challenge on arbitrary-shaped text-RRC-ArT," in *ICDAR*. IEEE, 2019, pp. 1571–1576.

[14] B. Shi, C. Yao, M. Liao *et al.*, "ICDAR2017 competition on reading Chinese text in the wild (RCTW-17)," in *14th IAPR ICDAR, ICDAR 2017, Kyoto, Japan, November 9-15, 2017*. IEEE, 2017, pp. 1429–1434.

[15] Y. Sun, D. Karatzas, C. S. Chan *et al.*, "ICDAR 2019 competition on large-scale street view text with partial labeling - RRC-LSVT," in *ICDAR*. IEEE, 2019, pp. 1557–1562.

[16] N. Nayef, C. Liu, J. Ogier *et al.*, "ICDAR2019 robust reading challenge on multi-lingual scene text detection and recognition - RRC-MLT-2019," in *ICDAR*. IEEE, 2019, pp. 1582–1587.

[17] T. Yuan, Z. Zhu, K. Xu *et al.*, "A large Chinese text dataset in the wild," *J. Comput. Sci. Technol.*, vol. 34, no. 3, pp. 509–521, 2019.

[18] B. Shi, X. Bai, and C. Yao, "Script identification in the wild via discriminative convolutional neural network," *Pattern Recognit.*, vol. 52, pp. 448–458, 2016.

[19] Z. Cao, J. Lu, S. Cui *et al.*, "Zero-shot handwritten Chinese character recognition with hierarchical decomposition embedding," *Pattern Recognit.*, vol. 107, p. 107488, 2020.

[20] C. Liu, F. Yin, D. Wang *et al.*, "CASIA online and offline Chinese handwriting databases," in *2011 ICDAR, ICDAR 2011, Beijing, China, September 18-21, 2011*. IEEE Computer Society, 2011, pp. 37–41.

[21] X. Ao, X. Zhang, H. Yang *et al.*, "Cross-modal prototype learning for zero-shot handwriting recognition," in *ICDAR*. IEEE, 2019, pp. 589–594.

[22] T. Wang, Z. Xie, Z. Li *et al.*, "Radical aggregation network for few-shot offline handwritten Chinese character recognition," *Pattern Recognit. Lett.*, vol. 125, pp. 821–827, 2019.

[23] J. Chen, B. Li, and X. Xue, "Zero-shot Chinese character recognition with stroke-level decomposition," in *IJCAI*. ijcai.org, 2021, pp. 615–621.

[24] C. Geng, S. Huang, and S. Chen, "Recent advances in open set recognition: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3614–3631, 2021.

[25] W. Chang, T. You, S. Seo *et al.*, "Domain-specific batch normalization for unsupervised domain adaptation," in *CVPR*. Computer Vision Foundation / IEEE, 2019, pp. 7354–7362.

[26] R. Yan, L. Peng, S. Xiao *et al.*, "Primitive representation learning for scene text recognition," in *CVPR*. Computer Vision Foundation / IEEE, 2021, pp. 284–293.

[27] A. K. Bhunia, A. Sain, A. Kumar *et al.*, "Joint visual semantic reasoning: Multi-stage decoder for text recognition," in *ICCV*. IEEE, 2021, pp. 14 920–14 929.
[28] H. Li, P. Wang, C. Shen *et al.*, "Show, attend and read: A simple and strong baseline for irregular text recognition," in *AAAI*. AAAI Press, 2019, pp. 8610–8617.
[29] C. Luo, L. Jin, and Z. Sun, "MORAN: A multi-object rectified attention network for scene text recognition," *Pattern Recognit.*, vol. 90, pp. 109–118, 2019.
[30] D. Yu, X. Li, C. Zhang *et al.*, "Towards accurate scene text recognition with semantic reasoning networks," in *CVPR*. IEEE, 2020, pp. 12 110–12 119.
[31] Y. Wang, H. Xie, S. Fang *et al.*, "From two to one: A new scene text recognizer with visual language modeling network," in *ICCV*. IEEE, 2021, pp. 14 174–14 183.
[32] T. DeVries, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.
[33] K. He, X. Chen, S. Xie *et al.*, "Masked autoencoders are scalable vision learners," in *CVPR*. IEEE, 2022, pp. 15 979–15 988.