Deep Radial Embedding for Visual Sequence Learning Supplementary Material

Yuecong Min[1,2], Peiqi Jiao[1,2], Yanan Li[3], Xiaotao Wang[3], Lei Lei[3],
Xiujuan Chai[4], and Xilin Chen[1,2]

[1] Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing, 100190, China
[2] University of Chinese Academy of Sciences, Beijing, 100049, China
[3] Xiaomi Inc., China
[4] Agricultural Information Institute, Chinese Academy of Agricultural Sciences, Beijing, 100081, China
{yuecong.min, peiqi.jiao}@vipl.ict.ac.cn, {liyanan3, wangxiaotao, leilei1}@xiaomi.com, chaixiujuan@caas.cn, xlchen@ict.ac.cn

This supplementary material provides details that are not shown in the main paper. We first present the implementation details on different datasets in Sect. A. Then we present the ablation results in Sect. B, including ablation results and more quantitative results. Finally, we present some unbalanced examples in Sect. C and show more visualization results in Sect. D.

A Implementation Details

A.1 Implementation Details on MNIST-Seq

We adopt the modified version of LeNet [7] as the *frame-wise feature extractor* (FFE) and set the dimension of output features to 3 for visualization. Adadelta [8] optimizer is used for 30-epoch training with an initial learning rate of 1.0. Each iteration processes 64 sequences and no augmentation technique is applied during training. For the hyperparameter choice, we adopt $\lambda_1 = 1.0$, $\beta = 0.2\pi$ and $\lambda_2 = 1.0$ as the default setting.

A.2 Implementation Details on Phoenix14

Following previous setting [4], we select ResNet18 [2] as the FFE. The gloss-wise temporal layer and two BiLSTM layers with 2×512 dimensional hidden states are adopted for temporal modeling. We adopt *Synchronized Cross-GPU Batch Normalization* (syncBN) [5] to gather statistics from all devices, which can accelerate the training process. Therefore, we shorten the training time and train all models for 40 epochs with a batch size of 2. Adam optimizer is used with an initial learning rate of 1e-4, which decays by a factor of 5 after epochs 25 and 35. The training set is augmented with random crop (224x224), horizontal flip (50%), and random temporal scaling (±20%). We replace the extra CTC (visual enhancement loss in [4]) with the proposed RadialCTC to show its effectiveness as intermediate supervision without using the visual alignment loss for simplicity. For the hyperparameter choice, we adopt $\lambda_1 = 1.0$, $\beta = 0.2\pi$ and $\lambda_2 = 0.1$ as the default setting.

A.3 Implementation Details on Scene Text Recognition

Following the setting of the benchmark, we adopt CRNN [6] as our baseline model. CRNN is optimized by CTC loss and has three components, including the convolution module, the recurrent module, and the decoder. The convolution module converts resized image ($1 \times 32 \times 100$) to a feature sequence of size $512 \times 1 \times 26$, where 512 is the dimension of output features. The recurrent layer has two BiLSTM layers with 2×256 hidden states and two fully-connected layers. It predicts a probability distribution among 37 predefined classes for each frame in the feature sequence. After that, the decoder converts the predictions into a label sequence. We train the model for 30 epochs with a batch size of 512 under the supervision of RadialCTC loss. Adam optimizer is used with β_1 set to 0.5 and an initial learning rate of 1e-3 decaying with a rate of 0.2 after epoch 10 and epoch 20. For the hyperparameter choice, we adopt $\lambda_1 = 1.0$, $\beta = 0.2\pi$ and $\lambda_2 = 0.1$ as the default setting.

A.4 Implementation Details on mAP Calculation

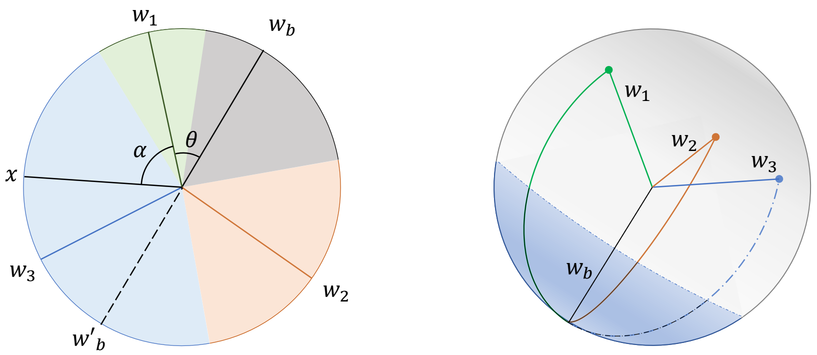
As the RadialCTC is designed to control the boundary of the recognized item, we calculate mAP like multi-label image recognition [1]. Specifically, we save the confidence for each frame and each class, and compute the precious/recall curve from the ranked confidence of each class. Recall R_n^k is defined as the proportion of frames belonging to the class k ranked above a given rank n. Precision P_n^k is the proportion of all frames above that rank that are from the given class k. The AP of the class k is defined as the mean precious at a set of 201 equally spaced recall levels $[0, 0.005, \cdots, 1]$: $AP_k = \sum_{n=1}^{200}(R_k^n - R_k^{n-1})P_k^n$. The final mAP is calculated by taking an average of all AP values per class $mAP = \frac{1}{K}\sum_{k=1}^{K} AP_k$. RadialCTC retains the iterative alignment mechanism of CTC, which ensures the continuity of predictions. Therefore, **a larger mAP** means more frames are recognized, which also indicates a larger overlap with the framewise annotation and **better localization accuracy**.

B Ablation Results

B.1 About the Choice of the Visualization Dimension

When ignoring the bias term in the classifier, no weight vector can satisfy the unique role of the blank class in two-dimensional space. Considering a special case with three non-blank classes shown in Fig 1a, the weight vectors of them are w_1, w_2 and w_3, respectively. If the weight vector of the blank w_b is between w_1 and w_2, we have:

$$\begin{aligned} & w_1^\intercal w_1 > w_b^\intercal w_1 \\ & \|w_1\| > \|w_b\| \, cos(\theta)' \end{aligned} \tag{1}$$



(a) Two-dimensional case

(b) Three-dimensional case

Fig. 1: Illustration of the visualization dimension. (a) hard to find a weight vector that can satisfy the role of the blank class in two-dimensional space; (b) an example weight vector of the blank class in three-dimensional space

where θ is the angle between w_1 and w_b. Then any vector between w_1 and w'_b will not be classified to the blank class:

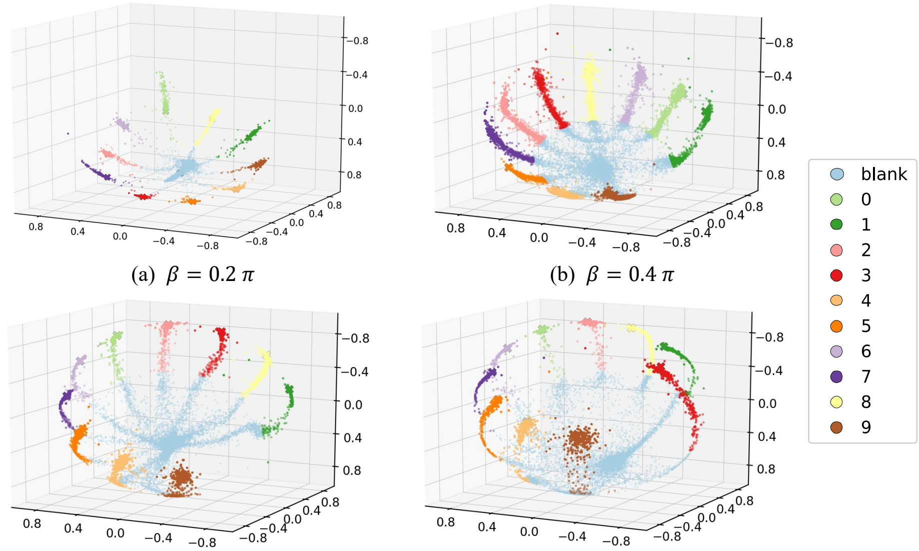
$$\begin{aligned} w_b^{\intercal}x &= \|w_b\|\,\|x\|\,cos(\alpha+\theta) \\ &= \|w_b\|\,\|x\|\,cos(\alpha)cos(\theta) - \|w_b\|\,\|x\|\,sin(\alpha)sin(\theta) \\ &< \|w_1\|\,\|x\|\,cos(\alpha) - \|w_b\|\,\|x\|\,sin(\alpha)sin(\theta) \\ &< w_1^{\intercal}x \end{aligned}. \tag{2}$$

Similarly, any vector between w_2 and w'_b will not be classified to the blank class. Therefore, we cannot find a weight vector of the blank class in two-dimensional space to satisfy that any frame between two non-blank keyframes can be classified into the blank class. This condition can be satisfied when the dimension of feature space is larger than two (e.g., Fig. 1b), so we choose the three-dimensional space for visualization.

B.2 Ablation Results on the Choice of Hyper-parameters

In Sect. 3, we propose a RadialCTC to constrain sequence features on a hypersphere while retaining the iterative alignment mechanism of CTC. To provide explanations of different constraints, we visualize the distribution of frame-wise features with different hyper-parameters (angle β in Fig. 2, weight of center regularization λ_2 in Fig. 3 and non-blank ratio η in Fig. 4) in the test set of Seq-MNIST.

Table 1 presents the ablation results of the angle hyper-parameter β on Seq-MNIST dataset ($\lambda_2 = 1.0$), and the corresponding distribution of features are visualized in Fig. 2. We can observe that the model achieves the best performance when the angle is 0.2π. We attribute this to the increase in arc length of distribution, which also increases the difficulty of classification.



(a) $\beta = 0.2\,\pi$ (b) $\beta = 0.4\,\pi$

(c) $\beta = 0.6\,\pi$ (d) $\beta = 0.8\,\pi$

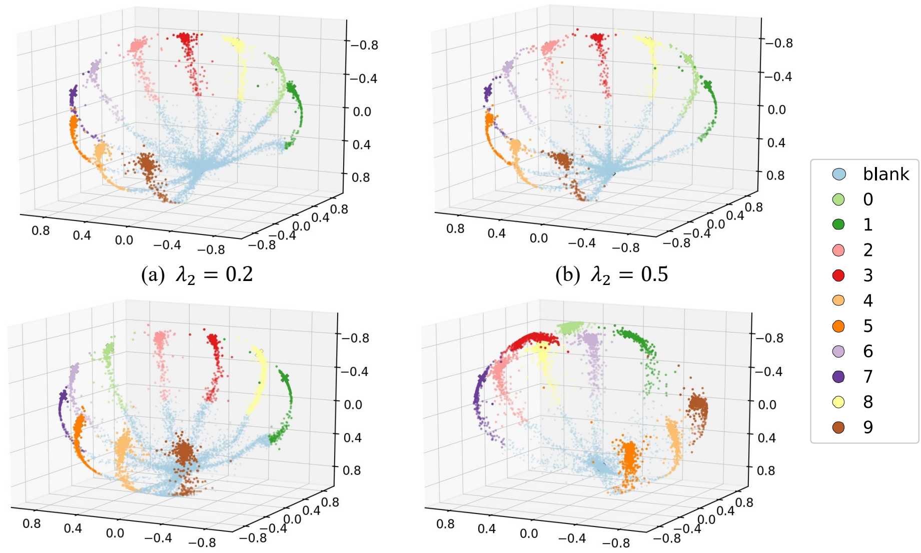
Fig. 2: The distribution of frame-wise features with different angle hyper-parameters β in test set of Seq-MNIST. Points with different colors are corresponding to different classes. **Best view in color**

Table 1: Recognition results (%) of the angle hyper-parameter β on Seq-MNIST

	β	0.2π	0.4π	0.6π	0.8π
Train	Acc.	99.6	99.2	97.9	97.0
Test	Acc.	**95.7**	94.8	94.8	94.3
	mAP	**23.4**	26.2	27.6	24.3

Ablation results on the weight λ_2 of center regularization are presented in Table 2 and Fig. 3 ($\beta = 0.6\pi$). The models with different settings achieve similar recognition results when the weight λ_2 is less than 1. The model tends to make more predictions with a large λ_2, but is also at risk of making more errors. Therefore, we adopt $\lambda_2 = 0.1$ on real-world datasets for better generalization.

As we mentioned in Sect. 3.3, we propose a non-blank ratio η to control the peaky behavior of CTC. As shown in Fig. 4, more frames are classified to non-blank classes as η increases, which validates that a large η can encourage the model to predict more non-blank labels. The improved localization results can also be observed from Table 3. Due to the unguaranteed quality of extended frames, the decision boundaries between non-blank classes look ambiguous when adopting large λ_2. Moreover, more frames tend to distribute near the decision boundary as η increases, which may affect the recognition results of the model. We have not found a proper solution that can take care of both localization and recognition results.



(a) $\lambda_2 = 0.2$

(b) $\lambda_2 = 0.5$

(c) $\lambda_2 = 1.0$

(d) $\lambda_2 = 10.0$

Fig. 3: The distribution of frame-wise features with different weights λ_2 of center regularization in test set of Seq-MNIST. Points with different colors are corresponding to different classes. **Best view in color**

Table 2: Recognition results (%) of the weight λ_2 on Seq-MNIST

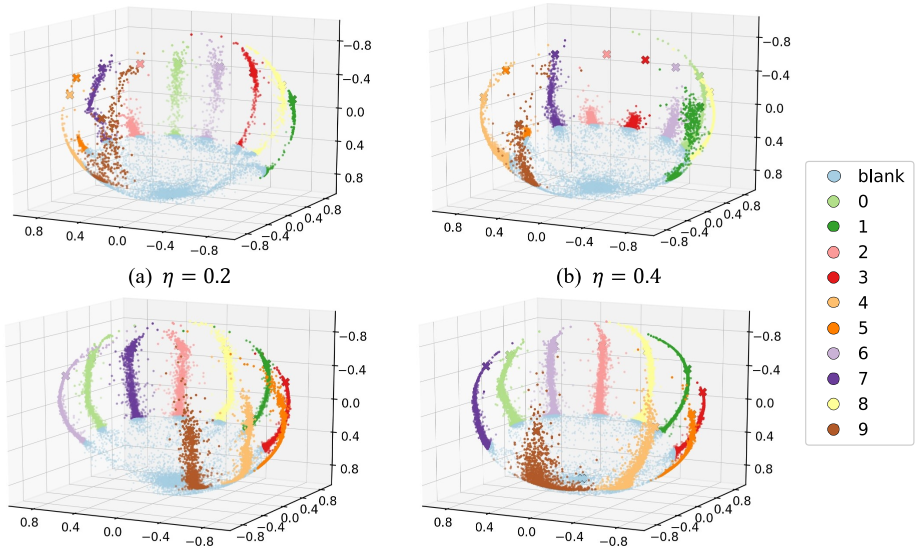
	λ_2	0.1	0.5	1.0	10.0
Train	Acc.	99.4	99.6	99.4	88.3
Test	Acc.	94.6	94.5	**94.8**	83.7
	mAP	24.0	24.6	**27.6**	39.1

B.3 Experimental Results with Three Dimensions

We present ablation studies on Seq-MNIST with high-dimensional output features in Sect. 4.2. Ablation results with higher dimensions on Seq-MNIST are present in Table 4, which have similar conclusions as Sect. 4.2. It is worth noting that adopting constraints can help the sequence model achieves better results when the dimension is high, which indicates these constraints are too strong to be helpful in the low-dimensional case. Besides, we have not found that improving localization ability is helpful for recognition. Although RadialCTC achieves competitive localization results under the unbalanced setting, it has much lower accuracy than in the balanced setting. We assume this is due to the different goals between recognition and localization, the former aims to extract discriminative features, and the latter pays more attention to low-level features.

B.4 More Quantitative Results on Phoenix14

Due to the ambiguous event boundaries in sequence data, the framewise annotation is costly and we have not found a proper dataset with frame-wise su-



(a) η = 0.2 (b) η = 0.4

(c) η = 0.6 (d) η = 0.8

Fig. 4: The distribution of frame-wise features with different non-blank ratios η in test set of Seq-MNIST. Points with different colors are corresponding to different classes. **Best view in color**

Table 3: Recognition results (%) of the non-blank ratio η on Seq-MNIST

	η	0.0	0.2	0.4	0.6	0.8
Train	Acc.	95.1	95.5	97.6	95.8	89.5
Test	Acc.	**90.3**	89.4	87.4	85.5	78.3
	mAP	21.2	42.1	59.3	74.6	**85.7**

pervision. Besides, we also conduct localization experiments on the real dataset (PHOENIX14) in Table 5, and the localization performance is based on frame-wise annotations estimated by HMM [3]. Experimental results show that RadialCTC can improve localization performance. We can also observe that RadialCTC does not bring significant performance gain for the predictions from the intermediate layer (Conv), but greatly improve the predictions from the primary layers (BiLSTM). This result suggests that radial embedding constrained by RadialCTC is helpful for sequence recognition.

CTC provides supervision via the Expectation-Maximization and easily reaches a local maximum, especially when adopting a powerful temporal model with limited training data. Previous work [4] has shown that adopting intermediate supervision can relieve the overfitting. As shown in Table 6, RadialCTC can further improve performance and relieve the overfitting with the proposed constraints in the low-resourced situation.

Table 4: Experimental results (%) on Seq-MNIST (dim=128)

Setting				balanced			unbalanced		
	Constraint			Train	Test		Train	Test	
Norm	Angle	Center	Radial	Acc.	Acc.	mAP	Acc.	Acc.	mAP
				99.9	96.4	25.5	99.8	**93.3**	25.7
✓				99.9	95.9	27.5	99.9	89.9	**35.8**
✓	✓			99.9	96.0	**34.7**	99.8	92.0	32.6
✓		✓		99.9	96.4	25.1	99.8	91.5	29.0
✓	✓	✓		99.9	**96.5**	27.2	99.9	91.1	27.6
✓	✓	✓	$\eta = 0.0$	98.0	94.3	21.5	98.3	84.1	21.2
✓	✓	✓	$\eta = 0.2$	99.6	**95.4**	40.7	99.2	**87.6**	42.8
✓	✓	✓	$\eta = 0.4$	97.4	91.0	66.6	95.4	82.6	66.3
✓	✓	✓	$\eta = 0.6$	97.4	89.4	82.8	81.9	75.9	81.9
✓	✓	✓	$\eta = 0.8$	90.6	81.9	**89.4**	67.3	69.4	**86.5**

Table 5: Experimental results (%) on Phoenix14 under different settings

Norm	Angle	Center	Radial	Train	Conv		BiLSTM	
				mAP	del/ins	WER	del/ins	WER
				37.1	8.4/4.2	**21.8**	7.3/2.6	21.0
✓				33.9	8.4/3.5	22.1	6.2/3.3	19.9
✓	✓			35.4	8.2/3.4	**21.8**	6.4/3.1	19.8
✓		✓		34.1	9.9/3.0	22.5	7.0/2.7	19.7
✓	✓	✓		34.6	8.8/3.1	**21.8**	6.5/2.7	**19.4**
✓	✓	✓	$\eta = 0.0$	32.4	10.2/2.7	23.2	6.2/2.6	19.6
✓	✓	✓	$\eta = 0.2$	39.6	7.9/3.8	**22.1**	6.3/2.7	**19.5**
✓	✓	✓	$\eta = 0.4$	44.6	8.1/5.1	24.1	6.3/3.0	20.3
✓	✓	✓	$\eta = 0.6$	47.4	7.6/6.9	24.9	7.0/2.8	20.1
✓	✓	✓	$\eta = 0.8$	**48.7**	5.8/10.8	29.3	7.7/2.7	21.2

C Limitation

In Sect. 4.2, we discuss the limitation of RadialCTC, here we provide several examples in Fig. 5. The first row is the weights of interpolation for each class, and remain rows provide several unbalanced examples of pseudo labeling on the training set of Seq-MNIST. Because RadialCTC controls the peaky behavior through a sequence-dependent term, it does not guarantee that each label has similar numbers of frames. For example, in the second row of Fig. 5, only 4 frames are recognized as 5 but 12 frames are recognized as 10.

Another concern is 'merged repeated labels problems' that the angular perturbation may choose to ignore the 'blank' between two 'non-blank' frames of the same label. RadialCTC provides a localization approach in a weakly-supervised manner, which also provides a solution that trains two models for recognition and localization, separately. A proper decoding approach (e.g., $\hat{\pi} = \arg\max_{\pi} p(\pi|\boldsymbol{v}, \hat{\boldsymbol{l}}; \theta)$) can decode more accurate localization results from the localization model based on the predictions $\hat{\boldsymbol{l}}$ of the recognition models, which

Table 6: Recognition results (%) on PHOENIX14 with sampled training data

	20%	40%	60%	80%
Baseline [26]	47.7/47.1	33.8/34.1	28.6/29.1	25.9/26.4
VAC [26]	35.3/35.1	27.4/28.2	24.3/25.1	22.6/23.5
RadialCTC	**33.4**/**32.8**	**26.2**/**26.4**	23.2/**23.2**	**21.0**/**21.6**
RadialCTC (η=0.2)	34.3/33.7	**26.2**/26.9	**22.8**/23.4	21.2/**21.6**

Table 7: Scene text recognition (%) with different non-blank ratios

	IIIT5K	SVT	IC03	IC13
$\eta = 0.2$	80.7	79.8	89.5	85.7
$\eta = 0.4$	80.2	79.6	89.4	85.2
$\eta = 0.6$	80.0	82.1	90.1	87.4
$\eta = 0.8$	76.6	74.0	85.2	82.0

can flexibly select hyperparameter η and can also solve the "merged repeated labels problem". Besides, RadialCTC aims to provide controllable boundaries and instance-wise localization beyond the goal of this submission. It will be an interesting research topic in the future.

Table 7 presents the relevant recognition results for scene text recognition datasets, this problem is only observed when adopting a large threshold (e.g., $\eta = 0.8$), which leads to performance degradation.

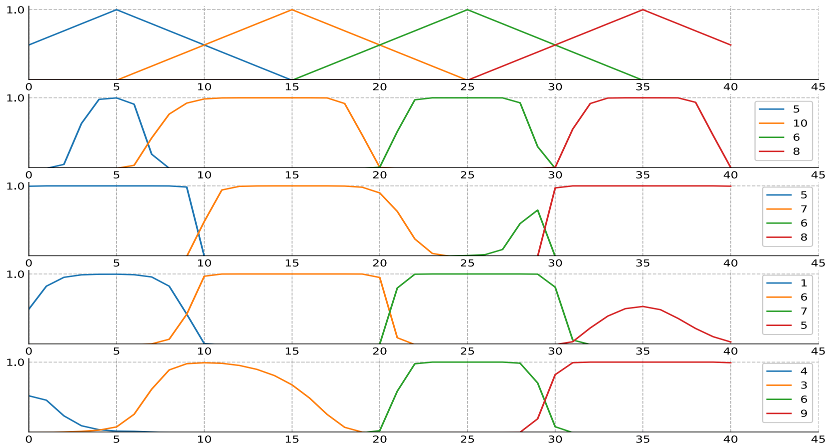
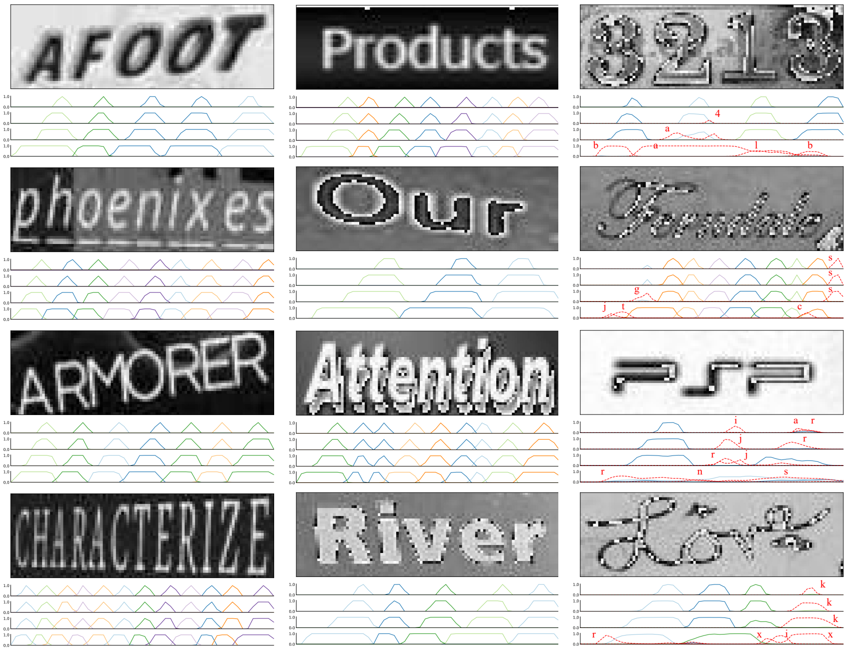


Fig. 5: Examples of unbalanced labelings. The top row presents the weights of interpolation for each class. The other rows present unbalanced pseudo labelings on the training set of Seq-MNIST-UB

D Visualization Results

More visualization results on scene text recognition are visualized in Fig. 6.



(a) Pseudo labels (b) Correct predictions (c) Wrong predictions

Fig. 6: Scene text recognition examples of pseudo labels and predictions with different non-blank ratios (η=0.2, 0.4, 0.6, 0.8 from top to bottom)

References

1. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88**(2), 303–338 (2010)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
3. Koller, O., Zargaran, S., Ney, H.: Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4297–4305 (2017)
4. Min, Y., Hao, A., Chai, X., Chen, X.: Visual alignment constraint for continuous sign language recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 11542–11551 (2021)
5. Peng, C., Xiao, T., Li, Z., Jiang, Y., Zhang, X., Jia, K., Yu, G., Sun, J.: Megdet: A large mini-batch object detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6181–6189 (2018)
6. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE transactions on pattern analysis and machine intelligence **39**(11), 2298–2304 (2016)
7. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: Proceedings of the European Conference on Computer Vision. pp. 499–515. Springer (2016)
8. Zeiler, M.D.: Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701 (2012)