Deep Radial Embedding for Visual Sequence Learning

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

Abstract. Connectionist Temporal Classification (CTC) is a popular objective function in sequence recognition, which provides supervision for unsegmented sequence data through aligning sequence and its corresponding labeling iteratively. The blank class of CTC plays a crucial role in the alignment process and is often considered responsible for the peaky behavior of CTC. In this study, we propose an objective function named RadialCTC that constrains sequence features on a hypersphere while retaining the iterative alignment mechanism of CTC. The learned features of each non-blank class are distributed on a radial arc from the center of the blank class, which provides a clear geometric interpretation and makes the alignment process more efficient. Besides, RadialCTC can control the peaky behavior by simply modifying the logit of the blank class. Experimental results of recognition and localization demonstrate the effectiveness of RadialCTC on two sequence recognition applications.

Keywords: Deep feature embedding · Visaul sequence learning · Sign language recognition · Scene text recognition

1 Introduction

Sequence data (*e.g.*, text, audio, and video) are present everywhere in daily life. Automatically analyzing and understanding sequences is a challenging yet fascinating field. As a fundamental task in sequence learning, sequence recognition aims to recognize occurred events from the data stream in a weakly supervised manner. Due to the continuity of the event, it is hard to identify its beginning and end points, which brings difficulties to both data annotation and analysis.

Recent years have witnessed the great success of deep learning in sequence learning tasks [2,37]. To achieve automatic alignment between sequence data and its corresponding labeling, Connectionist Temporal Classification (CTC) [10]

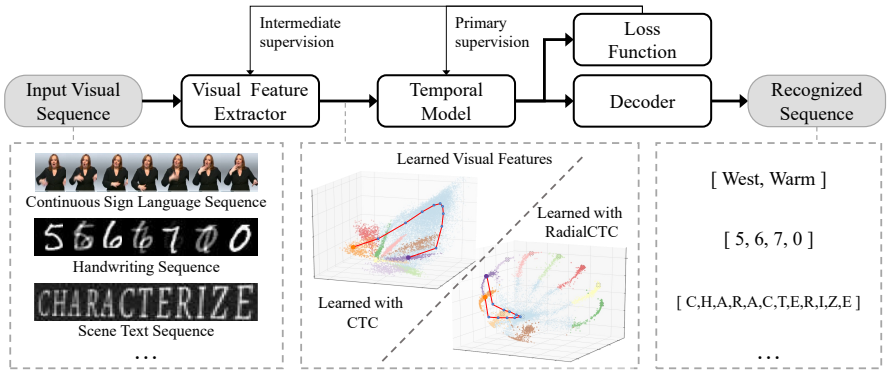

Fig. 1. The typical framework for visual sequence recongnition. **Best view in color**

leverages the natural monotonicity constraint that exists in sequence learning and adopts an extra 'blank' class to maximize the posterior probability of all feasible alignment paths. The typical framework is presented in Fig. 1, and CTC has been successfully adopted in many sequence learning applications [3,11,12,29] to provide intermediate or primary supervision.

One of the most interesting and controversial issues about CTC is its peaky behavior [10]: networks trained with CTC will conservatively predict a series of spikes. This peaky behavior helps the sequence recognition model quickly converge and fast decode. On the other hand, some works [4,23,43] regard the peaky behavior as a symptom of overfitting, which will deteriorate the performance dramatically when data are insufficient. Although CTC can achieve outstanding performance, its peaky behavior makes it unable to provide clear boundaries like some traditional statistical models (e.g., Hidden Markov Model) do.

Many works [13,14,22,23,43,44] provide interesting insights and possible solutions about the peaky behavior of CTC. Hu *et al.* [23] propose an entropy-based regularization term, which maximizes the entropy of feasible paths and penalizes the peaky distribution. Some works [14,22] try to improve the generalization ability of the model by extending the peaky prediction as frame-wise supervision. However, simply extending the peaky prediction may break the continuity of features and not take full advantage of CTC. Earlier works [13,43] show that CTC can be regarded as an iterative alignment process, which provides supervision via the Expectation-Maximization. Recent work [29] shows that training with CTC also makes feature norm peaky, which makes it easier to overfit when combined with powerful temporal models.

Inspired by the iterative alignment mechanism of CTC, we proposed an objective function named RadialCTC that constrains sequence features on a hypersphere. RadialCTC adopts several constraints and enforces the model to learn angularly discriminative features compared to the less constrained features in the inner space. As shown in Fig. 1, the proposed RadialCTC constrains features of non-blank classes to distribute on radial arcs from the center of the blank class.

Such a radial distribution provides a clear geometric interpretation of the peaky behavior of CTC and makes the alignment process more efficient.

Besides, the radial distribution of features provides an effective way to control the peaky behavior. Different from adding path-wise regularization [23] or modifying the peaky predictions [14,22], RadialCTC adopts a radial constraint to control the peaky behavior with the help of the iterative alignment mechanism of CTC. The radial constraint is implemented by simply adding an angular perturbation term on the blank logit. This term is dominated by a global non-blank ratio and sequence-wise angular distribution, providing consistent supervision for all sequence data. With the help of this constraint, RadialCTC can provide controllable event boundaries while achieving competitive recognition accuracy.

To show the effectiveness of RadialCTC, we conduct thoughtful experiments on a simulated sequence recognition dataset and two public benchmarks. Experimental results of recognition and localization demonstrate the effectiveness of RadialCTC. The major contributions are summarized as follows:

- Proposing the RadialCTC for sequence feature learning, which constrains sequence features on a hypersphere while retaining the iterative alignment mechanism of CTC. Features of non-blank classes are distributed on radial arcs from the center of the blank class.
- Proposing a simple angular perturbation term to control the peaky behavior, which can provide consistent supervision for all sequence data considering sequence-wise angular distribution.
- Conducting thoughtful experiments about the relationship between recognition and localization. Experimental results show the effectiveness of RadialCTC, which achieves competitive results on two sequence recognition applications and can also provide controllable event boundaries.

2 Related Work

2.1 Connectionist Temporal Classification

CTC [10] is proposed to provide supervision for unsegmented sequence data, which has shown advantages in many sequence recognition tasks (*e.g.* handwriting recognition [12], speech recognition [11], and sign language recognition [1,21,29]). Compared to other attention-based methods [2,37], CTC satisfies the monotonous nature of sequence recognition, and the CNN-LSTM-CTC model becomes a popular framework in sequence recognition tasks [5,35]. A controversial characteristic of CTC is its spike phenomenon [10]: networks trained with CTC will conservatively predict a series of spikes. The spike phenomenon can accelerate the decoding process but is also regarded as a symptom of overfitting [23,29]. Liu *et al.* [23] propose a entropy-based regularization method to penalize the peaky distribution and encourage exploration. Min *et al.* [29] propose a visual alignment constraint to enhance feature extraction before the powerful temporal module. Adding constraints on the CTC-based framework

can alleviate the overfitting problem. However, the peaky behavior still exists, and it is hard to provide clear event boundaries.

Many works [13,14,22,43,44] try to understand the peaky behavior of CTC. Earlier speech recognition works [13,43] interpret CTC as a special kind of Hidden Markov Model [33], which is trained with the Baum-Welch soft alignment algorithm, and the alignment result is updated at each iteration. Some recent works [22,14] leverage this iterative fitting characteristic and extend the spiky activations to get better recognition performance. However, these methods change the pseudo label at each iteration manually and may break the continuity of the sequence feature. Similar work to ours is [44], where the authors find that the peaky behavior is a property of local convergence, and the peaky behavior can be suboptimal. Different to [44], we constrain sequence features on a hypersphere and control the peaky behavior with an angular perturbation term.

2.2 Deep Feature Learning

The main goal of deep feature learning is to learn discriminative feature space with proper supervision. In some fine-grained image classification tasks (*e.g.*, face recognition), an important technical route is to learn strong discriminative features by improving the conventional softmax loss. Several margin-based losses [8,25,26,39] are proposed to learn more separable feature space. Wen *et al.* [41] propose to learn a center for each class and minimize the distance between deep features and their corresponding class centers, which can reduce intra-class variance. L-softmax [26] ignores the bias term in the classifier and adopts an angle-based margin to constrain the angles between learned features and their corresponding weights. SphereFace [25] further normalizes the weights of the classifier and constrains the learned feature on the unit hypersphere. On the other hand, several works [31,34] observe that the L_2-norm of the feature is informative to its quality and adopt the feature normalization to overcome sample distribution bias. Wang *et al.* [38] show the necessity of normalization and normalizing both features and weights, which become a common strategy in the following works [8,39]. Several angular-margin based losses [8,28,39] are proposed to further improve the recognition results.

3 Method

3.1 A Toy Sequence Recognition Example

To better illustrate the proposed method, we first build a simulated sequence recognition dataset named Seq-MNIST. Each sequence of Seq-MNIST is generated with four keyframes sampled from the MNIST database [19]. The transition clip from the former keyframe to the next is generated by interpolating α frames between them: the next keyframe fades in, and the former keyframe fades out. An example of the generation process is visualized in Fig. 2. The Seq-MNIST has 15,000 training sequences and 2500 testing sequences, and each sequence

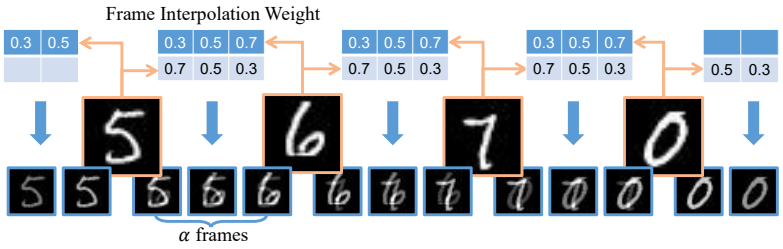

Fig. 2. An illustration of the data generation process of Seq-MNIST

contains 41 frames (α=9 and 5 additional transition frames at the beginning and the end). The proposed Seq-MNIST can be used to explore the design of the sequence recognition model, which is expected to recognize numbers from the generated sequence (*e.g.*, [5, 6, 7, 0] for the sequence in Fig. 2).

We adopt the modified version of LeNet [41] as the *frame-wise feature extractor* (FFE) and set the dimension of output features to 3 for visualization. The feature extractor takes the image sequence of T frames $\boldsymbol{x} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_T)$ and abstracts frame-wise features $\boldsymbol{v} = (\boldsymbol{v}_1, \cdots, \boldsymbol{v}_T)$. To clearly illustrate the relationship between frames, no temporal module is adopted in this sequence recognition model. The output features are fed to the *fully-connected* (FC) layer with n+1 output neurons (the vocubulary $\mathcal{V}$ contains n non-blank classes and one extra 'blank' class) for recognition. The whole process is formulated as:

$$\boldsymbol{v} = \mathrm{FFE}(\boldsymbol{x}), \quad y_i^t = \frac{e^{W_i^\intercal v_t + \boldsymbol{b}_i}}{\sum_{j=1}^{N} e^{W_j^\intercal v_t + \boldsymbol{b}_j}}, \tag{1}$$

where $\boldsymbol{v} \in \mathbb{R}^{T \times d}$ and d is the dimension of features. $W \in \mathbb{R}^{d \times (n+1)}$ and $\boldsymbol{b} \in \mathbb{R}^{n+1}$ are the weight matrix and the bias term of the FC layer.

As a widely-used loss function for weakly-supervised sequence recognition, CTC provides supervision by considering all possible alignments and maximizing the sum of their probabilities. With the help of an extra 'blank' class, CTC defines a many-to-one mapping $\mathcal{B} : \mathcal{V}^T \rightarrow \mathcal{V}^{\leq T}$ to align the alignment path π and its corresponding labeling $\boldsymbol{l}$. This mapping is achieved by successively removing the repeated labels and blanks in the path. For example, $\mathcal{B}$(-*aaa*--*aabbb*-) = $\mathcal{B}$(-*a*-*ab*-) = *aab*. The posterior probability of the labeling can be calculated by:

$$\begin{aligned} p(l|v) &= \sum_{\pi} p(\pi|\boldsymbol{l}, \boldsymbol{v}) = \sum_{\pi \in \mathcal{B}^{-1}(l)} p(\pi|\boldsymbol{v}) \\ p(\pi|\boldsymbol{v}) &= \prod_{t=1}^{T} p(\pi_t|\boldsymbol{v}) = \prod_{t=1}^{T} y_{\pi_t}^t \end{aligned} . \tag{2}$$

The frame-wise features $\boldsymbol{v}$ abstracted by the trained model are visualized in Fig. 3(a). Although we adopt a small feature dimension for visualization, it can reflect some characteristics of feature space and inspire us to optimize

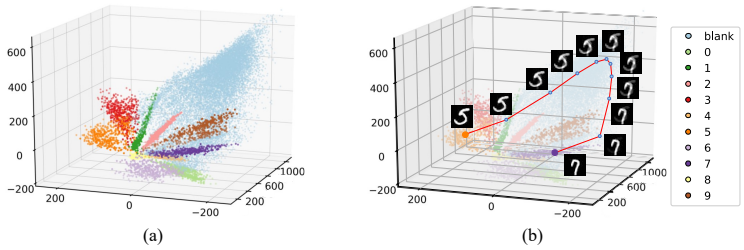

Fig. 3. Visualization of (a) the distribution of frame-wise features and (b) an example of transition trajectory in the test set of Seq-MNIST. Points with different colors are cooresponding to different classes. **Best view in color**

the design of loss function, which can also be extended to the high-dimensional case. From Fig. 3 we can observe that: (1) after training with CTC, the frame-wise features are separable among non-blank classes, but the decision boundary between non-blank classes and the blank class is pretty complicated, (2) over half of the features are classified to the blank class, which is corresponding to the peaky behavior of CTC, and features of the blank class have a large intra-class variance, and (3) although some transition frames are pretty similar to the keyframe, they are classified to the blank class as Fig. 3(b) shown.

3.2 The Design of Loss Function for Sequence Recognition

Many efforts have been devoted to learning discriminative features for fine-grained image classification, and we briefly summarize relevant designs as:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} \Bigg(\underbrace{-\log p(l_i|v_i, m)}_{\text{margin-based loss}} + \underbrace{\frac{1}{2}\left\|v_i - c_{l_i}\right\|_2}_{\text{center regularization}} \Bigg) \\ \text{s.t.} \quad & \underbrace{\tilde{v}_i = \frac{s v_i}{\left\|v_i\right\|_2}, \quad i = 1, \cdots, N, \quad \tilde{W}_j = \frac{W_j}{\left\|W_j\right\|_2}, \quad j = 1, \cdots, C}_{\text{normalization}} \end{aligned}, \tag{3}$$

where v_i and l_i are visual feature of sample i and its label, and c_k is the center vector of class k. s controls the feature scale to ensure the convergence. The center constraint aims to reduce intra-class variance [41], the normalization constraint can provide a geometric interpretation [25,38] and reduce the training data imbalance issue [25], and the margin-based loss [8,25,26,28,39] can enforce intra-class compactness and inter-class separation.

Like fine-grained image classification, sequence recognition needs a *discriminative* yet *steady* feature space. Inspired by the design of loss function in Equ. (3), we propose several constraints in the context of sequence learning.

Normalization. Different from image classification, sequence recognition not only needs to learn steady feature space but also needs to learn a generalized

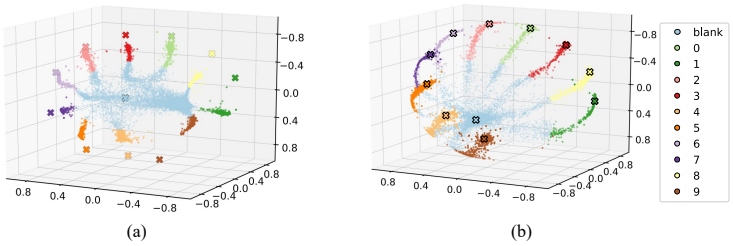

Fig. 4. The distribution of frame-wise features with (a) normalization and (b) normalization, angle and center constraints in the test set of Seq-MNIST, Points with different colors correspond to different classes. **Best view in color**

feature space from weakly supervised labeling. As shown in Fig. 3, the learned features with supervision from CTC have large intra-class variance, especially on the blank class, and the decision boundary between the blank class and non-blank classes is not clear. The vanilla CTC takes the inner distance between features and weights as input and provides little constraint on the alignment process. To learn a more separable feature space, we normalize both the features and weights and constrain the learned features on a hypersphere, which has been proven a practical approach in face recognition [25,38]. Fig. 4(a) shows the learned feature distribution after normalization. *After constraining all features on the hypersphere, the search space of the alignment process is reduced considerably, and features are distributed along several disjoint paths from the center of the blank class.* Besides, these features are not equally distributed among different classes and tend to distribute near the decision boundary rather than its class center. We further propose angle and center constraints to relieve these problems and make the features more discriminative.

Angle regularization. The blank class plays a unique role in CTC that the model trained with CTC will predict blank labels at uncertain frames. In other words, any frames between two non-blank keyframes can be classified into the blank class. Therefore, any transition trajectory between two non-blank keyframes will go through the decision region of the blank class as shown in Fig. 3(b). The data distribution and the recognition difficulty will affect the angle between the blank and non-blank classes. To enhance the discriminative and the generalization ability of the model, we propose an angle regularization term to minimize the distance between $\tilde{W}_b^{\intercal}\tilde{W}_{nb}$ and a given value $\cos(\beta)$.

Center regularization. As shown in Fig. 4(a), features are likely to be near the decision boundary of the blank class. CTC provides supervision by considering all possible alignments and has no explicit constraint on the separability of frame-wise features. Inspired by the pioneering work [38] that reduces intra-class variance by minimizing the distance between the deep feature and its corresponding class center, we assume that it is also helpful for sequence recognition. However, sequence recognition generally does not require frame-wise labels, and sequences often have many uncertain frames. Indiscriminately applying center

regularization on all frames will affect the representation steadiness and generalization ability of the model. Therefore, we only apply center regularization on keyframes set $KF(\boldsymbol{v})$, which is implemented by first estimating the alignment path $\hat{\pi} = \arg\max_{\pi} p(\pi|\boldsymbol{v}, \boldsymbol{l}; \theta)$ with the maximal probability as previous work [7] does and then minimizing the distance between features of keyframes in $\hat{\pi}$ and their corresponding classes.

Fig. 4(b) visualizes the learned feature distribution with the above constraints. These constraints provide a clear geometric interpretation for the sequence recognition with CTC supervision: the blank class plays a central role in the sequence recognition and the features of transition frames distributed on the disjoint arcs between centers of the blank class and non-blank classes.

The conservative supervision from CTC only classified a small ratio of frames to non-blank classes, but we can observe from Fig. 4(b) that features of the blank class are also clustered into several groups, which are distributed along the disjoint arcs to the centers of non-blank classes. This observation raises two questions: *(1) can we obtain accurate localization information from CTC, and (2) what is the relationship between the recognition and localization abilities of the model trained with CTC?*

The role of the angular margin. Adopting an angular/cosine-margin-based constraint is popular in deep feature learning, which can make learned features more discriminative by adding a margin term in softmax loss. Different from fully-supervised learning, sequence recognition does not require frame-wise annotation generally. It is hard to generate reliable frame-wise labels to apply a margin-based constraint on the sequence recognition model, but, *what will happen if we directly add an angular margin on a frame of the blank class?*

Several previous works [13,43] regard the CTC method as a special case of HMM, which is trained with Baum-Welch soft alignment at each iteration, and its optimization process can be interpreted via Expectation-Maximization. The gradient of $p(\boldsymbol{l}|\boldsymbol{x})$ with regard to the logit a_k^t [10] is:

$$\frac{\partial \ln p(\boldsymbol{l}|\boldsymbol{v})}{\partial a_k^t} = y_k^t - \hat{y}_k^t, \tag{4}$$

where $\hat{y}_k^t$ is the conditinal expected predictions calculated based on the *Forward-Backward* Algorithm (FB) [10]:

$$\hat{y}_k^t = FB(t, k, \boldsymbol{y}, \boldsymbol{l}) = \frac{1}{p(\boldsymbol{l}|\boldsymbol{v}; \theta)} \sum_{\pi \in \mathcal{B}^{-1}(\boldsymbol{l}), \pi_t = k} p(\pi|\boldsymbol{v}; \theta). \tag{5}$$

The frame-wise gradient of CTC has the same formulation as the Cross-Entropy (CE) loss, and the optimization of CTC is equivalent to iterative fitting [22]. However, the pseudo label $\hat{y}_k^t$ is calculated by considering probabilities of all feasible paths. In other words, changes in the logit also influence the probabilities of relevant paths, and *adding a margin term on one frame also changes the pseudo labels of its neighboring frames.*

Fig. 5 presents an example to illustrate this characteristic of CTC. For a sequence with five frames and labeling AB, the pseudo label $\hat{y}_k^t$ is calculated by

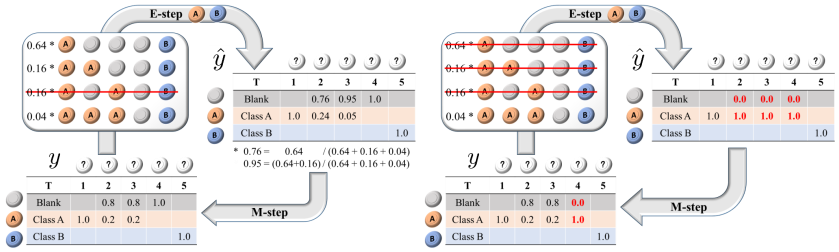

Fig. 5. An illstration of the iterative alignment mechanism of CTC

filtering out infeasible alignments and then calculating the expectation values. When adopting a large margin term in softmax operation to change the output of the fourth frame in Fig. 5(a) from (1.0, 0.0, 0.0) to (0.0, 1.0, 0.0), the pseudo labels of 2-4th frames changes from the blank class to class A. In this example, the angular margin plays a role in perturbation and provides a way to change the pseudo label while retaining the iterative alignment mechanism of CTC.

3.3 RadialCTC

Adopting an angular perturbation term can change the pseudo label, which provides a valuable tool to control the peaky behavior. However, it is hard to choose reliable frames to add this term, and a pre-defined term is hardly suitable for all sequences. Therefore, we try to control the peaky behavior of CTC by perturbing blank logits of all frames with a sequence-dependent term.

As the decision boundaries between the blank class and non-blank classes are similar, we look into the decision criteria of softmax in the binary case. After normalizing both features and weights and ignoring the bias term, the decision boundary between the blank class b and a non-blank class nb is $\theta_1 = \theta_2$, where $\theta_1 = \arccos(\tilde{W}_b^\intercal \tilde{v})$ and $\theta_2 = \arccos(\tilde{W}_{nb}^\intercal \tilde{v})$. A frame is recognized as the blank class when it lies on the hyperarc-like region ω_1 with $\theta_1 < \theta_2$, and Fig. 6 provides both 2D and 3D examples for better understanding. We can shrink the constrained region of the blank class from $\theta_1 < \theta_2$ to $\theta_1 + m < \theta_2$ by adding an angular perturbation term m on the blank frame. However, the prediction will soon become peaky again when training the model with this term and CTC because the learned features tend to evolve along with the decision boundary.

To control the peaky behavior flexibly, we propose a radial constraint that is implemented by adding an angular perturbation term $m(\eta, \boldsymbol{\theta}, \boldsymbol{l})$ between $\tilde{\boldsymbol{v}}$ and $\tilde{W}_b$ and adopt *the pseudo label of the perturbed logits* to provide supervision for *the original logits*. Unlike adopting a global perturbation term, we search for a proper frame within the sequence and move the decision boundary based on its feature to satisfy a pre-defined non-blank ratio η. Specially, given visual features $\boldsymbol{v} = (v_1, \cdots, v_T)$ and its corresponding labeling $\boldsymbol{l} = (l_1, \cdots, l_U)$, we find the frame v_τ which has the kth ($k = U + 1 + \lfloor (T - U) * \eta \rfloor$) largest angular

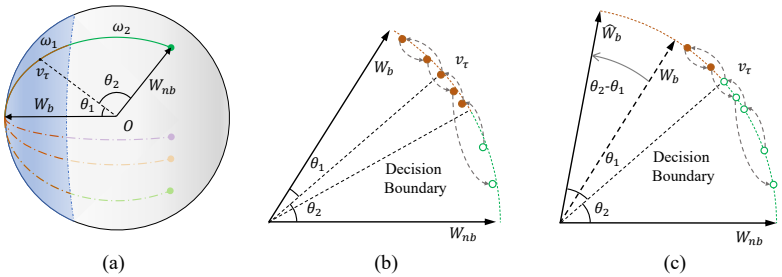

Fig. 6. The geometric interpretation of the angluar perturbation process. Visualization of the decision boundaries of (a) CTC on hypersphere, (b) CTC in binary case and (c) RadialCTC in binary case

difference between the blank class and the class with the highest probability that has appeared in the labeling. This process can be formulated as:

$$\begin{aligned} \boldsymbol{\theta} &= \arccos(\tilde{W}^\intercal \tilde{\boldsymbol{v}}) \\ m(\eta, \boldsymbol{\theta}, \boldsymbol{l}), \tau &= \text{topk}(\max_{c \in \boldsymbol{l}} \boldsymbol{\theta}_c - \boldsymbol{\theta}_b), \end{aligned} \tag{6}$$

where topk returns the kth largest value and its corresponding index. Then we add this angular perturbation term between $\tilde{\boldsymbol{v}}$ and $\tilde{W}_b$ and calculate the perturbed prediction $\boldsymbol{z}$, which is calculated by:

$$\boldsymbol{z}_i = \begin{cases} \dfrac{e^{s\cos(\boldsymbol{\theta}_b + m(\eta,\boldsymbol{\theta},\boldsymbol{l}))}}{e^{s\cos(\boldsymbol{\theta}_b + m(\eta,\boldsymbol{\theta},\boldsymbol{l}))} + \sum_{j=1, j\neq b}^{n+1} e^{s\cos(\boldsymbol{\theta}_j)}}, & \text{if } i = b \\ \dfrac{e^{s\cos(\boldsymbol{\theta}_i)}}{e^{s\cos(\boldsymbol{\theta}_b + m(\eta,\boldsymbol{\theta},\boldsymbol{l}))} + \sum_{j=1, j\neq b}^{n+1} e^{s\cos(\boldsymbol{\theta}_j)}}, & \text{otherwise} \end{cases}. \tag{7}$$

Fig. 6(b) and 6(c) provide the geometric interpretation of $m(\eta, \boldsymbol{\theta}, \boldsymbol{l})$ in the binary-class case. The original prediction of CTC is peaky, and only two frames are classified to label nb. To adjust the blank ratio, the 5th largest angular difference $(\theta_2 - \theta_1)$ is selected as $m(\eta, \boldsymbol{\theta}, \boldsymbol{l})$. The prediction calculation process of Equ. 7 is equivalent to rotating the weight vector of the blank class from W_b to the virtual weight vector $\hat{W}_b$. The decision boundary will move to the angular bisector of $\hat{W}_b$ and W_i and change the ratio of blanks as expected. The position of $\hat{W}_b$ is adjusted by the non-blank ratio η and the sequence-wise angular distribution, which is more flexible than a fixed global margin.

It seems like applying CTC to this modified prediction can adjust the ratio of non-blank supervision from CTC. However, the modified process leverages the labeling information, which is unknown during inference. A feasible solution is adopting the modified prediction as the pseudo label and iteratively narrowing the gap between it and the original prediction, which is similar to the iterative soft alignment mechanism of CTC. With the help of the FB Algorithm of CTC, we can calculate pseudo labels with higher quality and provide supervision for

the original prediction:

$$\begin{aligned} \hat{z}_k^t &= FB(t,k,\boldsymbol{z},\boldsymbol{l}) \\ L &= -\log p\Big(\boldsymbol{l}_i|\boldsymbol{v}_i, m(\eta,\boldsymbol{\theta},\boldsymbol{l})\Big) = -\sum_{t=1}^{T}\sum_{k=1}^{N}\hat{z}_k^t \log y_k^t \end{aligned} \tag{8}$$

The only difference between Equ. 8 and the original CTC (Equ.4 and Equ.5) is that the prediction used for pseudo label calculation is modified from the original prediction, which can flexibly adjust the blank ratio while retaining the iterative alignment mechanism of CTC. As the proposed method adjust the blank ratio based on 'radial' feature distribution, we named this method RadialCTC. The entire process can be formulated as:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N}\Bigg(\underbrace{-\log p\Big(\boldsymbol{l}_i|\boldsymbol{v}_i, m(\eta,\boldsymbol{\theta},\boldsymbol{l})\Big)}_{\text{radial constraint}} + \underbrace{\lambda_1 \sum_{j=1,j\neq b}^{C}\Big(\tilde{W}_b^{\intercal}\tilde{W}_j - \cos(\beta)\Big)^2}_{\text{angle regularization}} \\ & \underbrace{+\lambda_2 \sum_{t_i\in KF(\boldsymbol{v})}\Big\|\tilde{\boldsymbol{v}}_{t_i} - \tilde{W}_{y_{t_i}}\Big\|_2}_{\text{center regularization}}\Bigg) \\ \text{s.t.} \quad & \underbrace{\tilde{\boldsymbol{v}}_i = \frac{s\boldsymbol{v}_i}{\|\boldsymbol{v}_i\|_2}, \quad i=1,\cdots,N, \quad \tilde{W}_j = \frac{W_j}{\|W_j\|_2}, \quad j=1,\cdots,C}_{\text{normalization}} \end{aligned} \tag{9}$$

where λ_1 and λ_2 are hyperparameters to control the strength of regularization.

4 Experiments

This section conducts ablation studies on the Seq-MNIST and evaluates the recognition and localization results. To show the generalization ability of the proposed method, we exemplify it for sequence recognition with two applications: *continuous sign language recognition* (CSLR) and scene text recognition.

4.1 Datasets

Seq-MNIST. Seq-MNIST maintains the distribution balance from MNIST [19]. We also simulate an unbalanced training set by sampling images at the rate of 0.1 for classes 0 to 4 and remaining unchanged for others.

Phoenix14. As a popular CSLR dataset, Phoenix14 [17] contains about 12.5 hours of video data collected from weather forecast broadcast and is divided into three parts: 5,672 sentence for training, 540 for *development* (Dev), and 629 for *testing* (Test). It also provides a signer-independent setting where data of 8 signers are chosen for training and leave out data of signer05 for evaluation.

Table 1. Experimental results (%) on Seq-MNIST (dim=3)

Setting				Balanced			Unbalanced		
Constraint				Train	Test		Train	Test	
Norm	Angle	Center	Radial	Acc.	Acc.	mAP	Acc.	Acc.	mAP
				99.5	95.1	**30.7**	99.9	**87.8**	33.0
✓				99.6	94.8	24.6	98.8	79.5	42.7
✓	✓			99.5	95.2	25.9	98.7	79.5	36.1
✓		✓		99.7	**95.6**	26.9	98.1	74.2	**44.8**
✓	✓	✓		98.5	93.6	25.6	98.2	73.6	42.2
✓	✓	✓	$\eta = 0.0$	96.7	90.4	22.7	98.9	**81.5**	19.7
✓	✓	✓	$\eta = 0.2$	99.8	**93.7**	40.5	97.3	72.3	41.5
✓	✓	✓	$\eta = 0.4$	97.9	88.7	62.8	86.1	31.4	50.7
✓	✓	✓	$\eta = 0.6$	96.6	84.2	78.7	75.7	22.7	56.7
✓	✓	✓	$\eta = 0.8$	94.5	80.8	**87.9**	61.3	23.4	**60.9**

Scene Text Recognition Datasets. Following the standard experimental setting, we use the synthetic Synth90k [15] as training data and test our methods on four real-world benchmarks (*ICDAR-2003*(IC03) [27], *ICDAR-2013*(IC13) [16], *IIIT5k-word*(IIIT5k) [30] and *Street View Text*(SVT) [40]) without fine-tuning.

For Seq-MNIST and scene text recognition datasets, we use sequence accuracy as the evaluation metric. *Word error rate* (WER) is adopted as the evaluation metric of CSLR as previous work does [29]. We adopt the *mean Average Precious* (mAP) [9] to evaluate the localization performance. Other implementation details can be found in the Supplementary.

4.2 Experimental Results

Ablation on RadialCTC design. We adopt the modified LeNet [41] mentioned in Sect. 3.1 as baseline and present ablation results of RadialCTC on Seq-MNIST in Table 1. To better illustrate the effect of different constraints, we set the dimension of output features to 3. We can observe that adopting either angle or center regularization can improve the recognition results. However, the combined use of them leads to a performance drop. We assume this is because this constraint is too strong to learn separable features in this low-dimensional space, and it achieves better performance when the dimension increases to 128. We can also observe that RadialCTC performs worse on the extremely unbalanced setting, which tends to make more predictions (mAP increases from 33.0% to 42.2%) but also brings more errors. Visualization of learned features and more results can be found in the Supplementary.

Localization Ability of RadialCTC. We adopt the class with a larger interpolation weight as its label for each frame in the sequence and calculate the mAP of different settings to show their localization ability. As η increases, the localization performance significantly improves (from 22.7% to 87.9%) in Table 1. Although the sequence accuracy drops as η increases, we find this mainly because model trained with larger η tends to merge repeated labels into one

Table 2. Performance comparison (%) on Phoenix14 dataset under multi-signer setting

	Dev		Test	
	del/ins	WER	del/ins	WER
Re-Sign [18]	-	27.1	-	26.8
DNF [6]	7.8/3.5	23.8	7.8/3.4	24.4
CMA [32]	7.3/2.7	21.3	7.3/2.4	21.9
STMC [45]	7.7/3.4	21.1	7.4/2.6	20.7
SMKD [14]	6.8/**2.5**	20.8	6.3/**2.3**	21.0
Baseline [29]	7.3/2.6	21.0	7.6/3.0	22.6
RadialCTC	**6.5**/2.7	**19.4**	**6.1**/2.6	**20.2**

Table 3. Performance comparison (%) on Phoenix14 dataset under signer-independent setting

	Dev		Test	
	del/ins	WER	del/ins	WER
Re-Sign [18]	-	45.1	-	44.1
DNF [6]	**9.2**/4.3	36.0	**9.5**/4.6	35.7
CMA [32]	11.1/**2.4**	34.8	11.4/3.3	34.3
Baseline [29]	11.6/3.6	36.7	9.8/3.5	33.8
RadialCTC	10.5/2.9	**33.8**	9.7/**2.9**	**32.2**

Table 4. Performance comparison (%) on scene text recognition datasets

Method	IIIT5K	SVT	IC03	IC13
R2AM [20]	78.4	80.7	88.7	90.0
STAR-Net [24]	**83.3**	**83.6**	89.9	89.1
RARE [36]	81.9	81.9	90.1	88.6
CRNN [35]	78.2	80.8	89.4	86.7
EnEsCTC [23]	82.0	80.6	92.0	90.6
ACE(1D, Cross Entropy) [42]	82.3	82.6	92.1	89.7
Reinterpreting CTC [22]	81.1	82.2	91.2	87.7
Baseline [35]	79.8	80.4	89.9	87.3
RadialCTC	83.2	82.1	**92.3**	**90.7**

(e.g., predicting 567 rather than 5567). After ignoring this case, the recognition accuracy of the $\eta = 0.8$ setting improves from 80.8% to 94.6%, which indicates the proposed RadialCTC can provide accurate boundaries while achieving competitive recognition results. Besides, we can conclude that the localization ability of vanilla CTC is between $\eta = 0.0$ and $\eta = 0.2$ settings.

Limitations of RadialCTC. Experimental results on the unbalanced setting in Table 1 show some limitations of the RadialCTC. When the dataset is extremely unbalanced, prematurely adding center regularization will reduce intra-class variance before the class centers are sufficiently separable and damage the generalization ability of the model. Besides, the distribution of data also affects the localization ability of the model. RadialCTC controls the peaky behavior through a sequence-dependent term. Therefore, the model is more likely to predict the dominant sequence class and limits its localization ability on classes with fewer samples. This conclusion is reflected in the slower growth of mAP.

Continuous Sign Language Recognition. As a typical visual sequence recognition task, visual features play an essential role in CSLR. Recent work [29] has shown that adding an extra CTC on visual features can efficiently improve the recognition results. Therefore, we replace this extra CTC with RadialCTC to show its effectiveness as intermediate supervision without using the distillation loss for simplicity. As shown in Table 2, the proposed RadialCTC improves the recognition results and achieves sota results, which indicates that RadialCTC

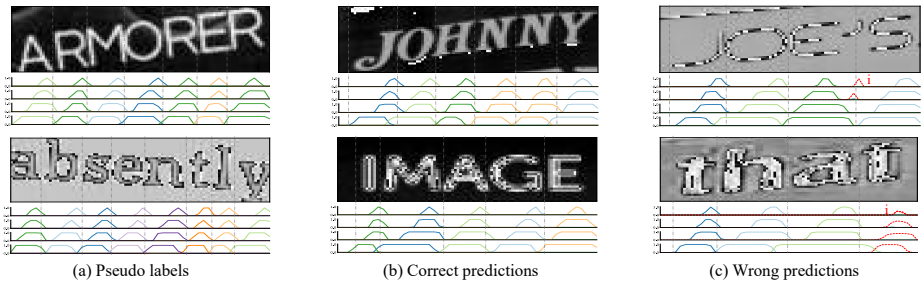

Fig. 7. Scene text recognition examples with different non-blank ratios (η=0.2, 0.4, 0.6, 0.8 from top to bottom)

can learn more discriminative features even if the dataset has limited data. Similar conclusions can be drawn from the signer-independent setting in Table 3.

Scene Text Recognition. As a classical framework for scene text recognition, modified versions of CTC [22,23,42] adopt CRNN [35] as the baseline and evaluate the performance on a standard benchmark. We follow this experimental setting and present results in Table 4. By changing the primary supervision from CTC to the proposed RadialCTC, the scene text recognition model achieves more than 1.7% improvement on four real-world test sets. The proposed method outperforms other modified versions of CTC on almost all test sets and is competitive with other methods. Previous works [22,23] are also driven by designing a proper method to relieve the peaky behavior. However, we have not found that improving localization ability is helpful for recognition, and the performance gain is obtained without using the radial constraint.

To better show the localization ability of RadialCTC, we visualize pseudo labels and predictions of different non-blank ratios in Fig. 7. RadialCTC can provide confident and accurate pseudo labels and predictions as η increases.

5 Conclusion

As a popular objective function in sequence recognition, CTC provides supervision for unsegmented sequences through an iterative alignment mechanism. In this study, we propose a RadialCTC that constrains sequence features on a hypersphere while retaining the iterative alignment mechanism of CTC. RadialCTC provides a clear geometric interpretation of the distribution of sequence features. Besides, an efficient constraint is proposed to control the peaky behavior of vanilla CTC. Experimental results show that RadialCTC can effectively improve recognition results and provide reliable localization results. We hope the proposed RadialCTC can be a useful tool in sequence recognition, and the geometric interpretation of CTC can inspire other sequence learning tasks.

Acknowledgement. This study was partially supported by the Natural Science Foundation of China under contract No.61976219.

References

1. Albanie, S., Varol, G., Momeni, L., Afouras, T., Chung, J.S., Fox, N., Zisserman, A.: Bsl-1k: Scaling up co-articulated sign language recognition using mouthing cues. In: European Conference on Computer Vision. pp. 35–53. Springer (2020)
2. Bahdanau, D., Cho, K.H., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: International Conference on Learning Representations (2015)
3. Camgoz, N.C., Hadfield, S., Koller, O., Bowden, R.: Subunets: End-to-end hand shape and continuous sign language recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3075–3084 (2017)
4. Cheng, K.L., Yang, Z., Chen, Q., Tai, Y.W.: Fully convolutional networks for continuous sign language recognition. In: Proceedings of the European Conference on Computer Vision. pp. 697–714. Springer (2020)
5. Cihan Camgoz, N., Hadfield, S., Koller, O., Bowden, R.: Subunets: End-to-end hand shape and continuous sign language recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3056–3065 (2017)
6. Cui, R., Liu, H., Zhang, C.: Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7361–7369 (2017)
7. Cui, R., Liu, H., Zhang, C.: A deep neural framework for continuous sign language recognition by iterative training. IEEE Transactions on Multimedia **21**(7), 1880–1891 (2019)
8. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4690–4699 (2019)
9. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88**(2), 303–338 (2010)
10. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of International Conference on Machine Learning. pp. 369–376 (2006)
11. Graves, A., Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks. In: Proceedings of the International Conference on Machine Learning. pp. 1764–1772. PMLR (2014)
12. Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., Schmidhuber, J.: A novel connectionist system for unconstrained handwriting recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(5), 855–868 (2008)
13. Hadian, H., Sameti, H., Povey, D., Khudanpur, S.: End-to-end speech recognition using lattice-free mmi. In: Interspeech. pp. 12–16 (2018)
14. Hao, A., Min, Y., Chen, X.: Self-mutual distillation learning for continuous sign language recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 11303–11312 (2021)
15. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. arXiv preprint arXiv:1406.2227 (2014)
16. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: Icdar 2013 robust reading competition. In: 2013 12th International Conference on Document Analysis and Recognition. pp. 1484–1493. IEEE (2013)

17. Koller, O., Forster, J., Ney, H.: Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. Computer Vision and Image Understanding **141**, 108–125 (2015)
18. 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)
19. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
20. Lee, C.Y., Osindero, S.: Recursive recurrent nets with attention modeling for ocr in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2231–2239 (2016)
21. Li, D., Rodriguez, C., Yu, X., Li, H.: Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision. pp. 1459–1469 (2020)
22. Li, H., Wang, W.: Reinterpreting ctc training as iterative fitting. Pattern Recognition **105**, 107392 (2020)
23. Liu, H., Jin, S., Zhang, C.: Connectionist temporal classification with maximum entropy regularization. Advances in Neural Information Processing Systems **31**, 831–841 (2018)
24. Liu, W., Chen, C., Wong, K.Y.K., Su, Z., Han, J.: Star-net: a spatial attention residue network for scene text recognition. In: Proceedings of the British Machine Vision Conference. vol. 2, p. 7 (2016)
25. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 212–220 (2017)
26. Liu, W., Wen, Y., Yu, Z., Yang, M.: Large-margin softmax loss for convolutional neural networks. In: Proceedings of the International Conference on Machine Learning. vol. 2, p. 7 (2016)
27. Lucas, S.M., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R., Ashida, K., Nagai, H., Okamoto, M., Yamamoto, H., et al.: Icdar 2003 robust reading competitions: entries, results, and future directions. International Journal of Document Analysis and Recognition **7**(2), 105–122 (2005)
28. Meng, Q., Zhao, S., Huang, Z., Zhou, F.: Magface: A universal representation for face recognition and quality assessment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 14225–14234 (2021)
29. 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)
30. Mishra, A., Alahari, K., Jawahar, C.: Scene text recognition using higher order language priors. In: Proceedings of the British Machine Vision Conference (2012)
31. Parde, C.J., Castillo, C., Hill, M.Q., Colon, Y.I., Sankaranarayanan, S., Chen, J.C., O'Toole, A.J.: Deep convolutional neural network features and the original image. arXiv preprint arXiv:1611.01751 (2016)
32. Pu, J., Zhou, W., Hu, H., Li, H.: Boosting continuous sign language recognition via cross modality augmentation. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 1497–1505 (2020)
33. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)
34. Ranjan, R., Castillo, C.D., Chellappa, R.: L2-constrained softmax loss for discriminative face verification. arXiv preprint arXiv:1703.09507 (2017)

35. 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)
36. Shi, B., Wang, X., Lyu, P., Yao, C., Bai, X.: Robust scene text recognition with automatic rectification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4168–4176 (2016)
37. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in Neural Information Processing Systems **30** (2017)
38. Wang, F., Xiang, X., Cheng, J., Yuille, A.L.: Normface: L2 hypersphere embedding for face verification. In: Proceedings of the 25th ACM International Conference on Multimedia. pp. 1041–1049 (2017)
39. Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., Liu, W.: Cosface: Large margin cosine loss for deep face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5265–5274 (2018)
40. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1457–1464. IEEE (2011)
41. 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)
42. Xie, Z., Huang, Y., Zhu, Y., Jin, L., Liu, Y., Xie, L.: Aggregation cross-entropy for sequence recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6538–6547 (2019)
43. Zeyer, A., Beck, E., Schlüter, R., Ney, H.: Ctc in the context of generalized full-sum hmm training. In: Interspeech. pp. 944–948 (2017)
44. Zeyer, A., Schlüter, R., Ney, H.: Why does ctc result in peaky behavior? arXiv preprint arXiv:2105.14849 (2021)
45. Zhou, H., Zhou, W., Zhou, Y., Li, H.: Spatial-temporal multi-cue network for continuous sign language recognition. In: Proceedings of the Association for the Advancement of Artificial Intelligence. pp. 13009–13016 (2020)