Cross-Sentence Gloss Consistency for Continuous Sign Language Recognition

Qi Rao[1]*, Ke Sun[2], Xiaohan Wang[3], Qi Wang[2], Bang Zhang[2]

[1]ReLER, AAII, University of Technology Sydney
[2]Institute for Intelligent Computing, Alibaba Group
[3]Stanford University

Abstract

Continuous sign language recognition (CSLR) aims to recognize gloss sequences from continuous sign videos. Recent works enhance the gloss representation consistency by mining correlations between visual and contextual modules within individual sentences. However, there still remain much richer correlations among glosses across different sentences. In this paper, we present a simple yet effective Cross-Sentence Gloss Consistency (CSGC), which enforces glosses belonging to a same category to be more consistent in representation than those belonging to different categories, across all training sentences. Specifically, in CSGC, a prototype is maintained for each gloss category and benefits the gloss discrimination in a contrastive way. Thanks to the well-distinguished gloss prototype, an auxiliary similarity classifier is devised to enhance the recognition clues, thus yielding more accurate results. Extensive experiments conducted on three CSLR datasets show that our proposed CSGC significantly boosts the performance of CSLR, surpassing existing state-of-the-art works by large margins (*i.e.*, 1.6% on PHOENIX14, 2.4% on PHOENIX14-T, and 5.7% on CSL-Daily).

Introduction

Automatically recognizing signs from videos is significant for communication among the deaf and hard-of-hearing community. The Continuous Sign Language Recognition (CSLR) task is designed to recognize glosses (*i.e.*, basic semantic symbols associated with signs) from a continuous video stream.

Current CSLR models (Zhou et al. 2020; Min et al. 2021; Hao, Min, and Chen 2021; Zuo and Mak 2022; Hu et al. 2022) typically consist of two modules for feature extraction: a visual module to extract short-term spatial-temporal information from the input frames and followed by a contextual (sequential) module aggregating long-term contextual information. An alignment module is utilized to align the extracted features with corresponding gloss labels. According to (Zuo and Mak 2022), recent methods improve CSLR by enhancing the consistency between visual and contextual modules, which is measured at a frame level (Min et al. 2021; Hao, Min, and Chen 2021) or sentence level (Zuo and Mak 2022). However, their consistency are all limited in a single sentence scope. The lack of adequate gloss correlations hinders its representation learning. In contrast, the rich gloss correlations across different sentences, as an intuitive sense to recognize language signs, has not been studied.

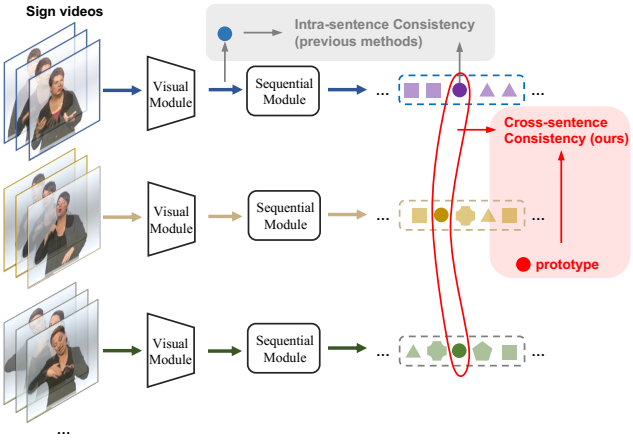

Figure 1: Previous CSLR methods constrain the representation consistency between visual and contextual modules only within the same sentence scope, i.e., intra-sentence consistency, yet ignoring the rich correlations among cross-sentence gloss features in the dataset scope. We reckon gloss features belonging to a same category across different sentences are expected to be consistent in representation, thus presenting our Cross-Sentence Gloss Consistency.

To investigate the above gloss correlations across sentences, in this paper, we study the impact of cross-sentence consistency for CSLR. As illustrated in Fig. 1, limiting the representation consistency within individual sentences is hard to distill comprehensive knowledge for gloss discrimination. Based on this observation, we present Cross-Sentence Gloss Consistency (CSGC) for CSLR. Instead of previous intra-sentence consistency, our CSGC exploits the cross-sentence gloss correlations in a global (dataset) scope, thus efficiently benefiting the learning of gloss discrimination and significantly boosting the CSLR performance.

*Work done during an internship at Alibaba Group.

Specifically, CSGC is constructed based on a core gloss category representative component: Gloss Prototype (GP). GP is designed as a referral dictionary containing a set of category prototypes. Each gloss category is assigned with a prototype and is maintained constantly in the training stage. We fuse each sample of gloss feature into GP by momentum updating the corresponding prototype. In the meantime, GP benefits the discrimination of gloss samples in a contrastive way: on the one hand closing the gloss-to-prototype distance between the gloss sample and its corresponding category prototype, on the other hand enlarging the gloss-to-prototype distances between the gloss sample and other category prototypes. As the training progresses, GP is gradually comprehended with representations of all gloss samples, which is able to comprehensively describe each gloss category, leveraging exhaustive gloss knowledge learned among gloss correlations on the whole dataset.

Besides, considering the well-distinguished GP to bring benefits of our cross-sentence gloss consistency into the final recognition, we design a fusion module named Auxiliary Similarity Fusion Strategy (ASFS). ASFS enriches the initial recognition (*i.e.*, outputs of the fully-connected classifier) with cross-sentence gloss consistency clues. In ASFS, we first measure each gloss sample with a similarity table, *i.e.*, the gloss-to-prototype similarities with all prototypes in GP. Then by normalizing the similarity table, we obtain a gloss prediction probability in the aspect of cross-sentence gloss discrimination. Next fuse it with the output probability from the initial classifier (which contains only intra-sentence gloss discrimination) as our final recognition probability. Since ASFS considers the gloss prediction in both cross-sentence and intra-sentence perspectives, it further improves the recognition performance.

We evaluate our CSGC on three CSLR benchmarks, including PHOENIX14 (Koller, Forster, and Ney 2015), PHOENIX14-T (Camgoz et al. 2018) and CSL-Daily (Zhou et al. 2021). Quantitative results indicate that we surpass existing state-of-the-art works by big margins. Remarkably, we improve the state-of-the-art performance by 1.6% on PHOENIX14, 2.4% on PHOENIX14-T, and 5.7% on CSL-Daily.

Related Work

Continuous Sign Language Recognition

Feature representation is always an important part in CSLR. Before deep learning era, traditional methods (Freeman and Roth 1995; Gao et al. 2004; Han, Awad, and Sutherland 2009; Koller, Forster, and Ney 2015) usually devise handcrafted features to represent visual and temporal information. Bringing deep learning techniques into CSLR achieves great success (Koller et al. 2016; Koller, Zargaran, and Ney 2017) thanks to the significant feature representation ability of convolutional neural networks, recurrent neural networks and recent Transformers (Vaswani et al. 2017; Camgoz et al. 2020; Zuo and Mak 2022). Current CSLR methods follow a standard pipeline, which consists of two encoding backbones and an alignment module: a visual encoding backbone (*i.e.*, 2D-CNN (Zhou et al. 2020; Cheng et al. 2020; Min et al. 2021) and 3D-CNN (Pu, Zhou, and Li 2019)), a contextual encoding backbone including a short-term temporal modeling (*i.e.*, TCN (Min et al. 2021; Hu et al. 2022), temporal lifting (Hu et al. 2022)) and a long-term sequence modeling (*e.g.*, BiLSTM (Cui, Liu, and Zhang 2017; Min et al. 2021; Hu et al. 2022), Transformer (Hu et al. 2022; Niu and Mak 2020; Camgoz et al. 2020)) and a CTC loss (Graves et al. 2006) for alignment between the extracted gloss features and corresponding labels.

Particularly, a branch of works are target at enhancing the representation consistency. VAC (Min et al. 2021) proposes to constrain the alignment between visual and sequential outputs directly. SMKD (Hao, Min, and Chen 2021) balances the focus between short-term and long-term information from visual and contextual module by a shared classifier. C^2SLR (Zuo and Mak 2022) proposes a sentence embedding consistency between visual and contextual module. As a critical factor in gloss discrimination, the research of feature representation consistency in CSLR has been continuous for years.

However, these works mostly learn to understand signs under visual and contextual consistency within individual sentences while omitting the rich gloss correlations across different sentences. In contrast, our proposed cross-sentence gloss consistency fully leverages the inherent gloss discrimination, thus leading to more effective representation learning.

Prototype Learning

Previous cognitive psychological studies (Aamodt and Plaza 1994; Newell, Simon et al. 1972; Yang, Zhuang, and Pan 2021) indicate that people use past cases as models when learning to solve problems. In machine learning practice, the prototype based classification (Duda, Hart, and Stork 1973; Hastie et al. 2009; Shawe-Taylor, Cristianini et al. 2004) has been studied for long time, from traditional statistics approaches to Support Vector Machine to Multilayer Perceptions. Prototype based classification has experienced sustainable development thanks to its simple and intuitive insight: observations are directly compared with representative prototypes. Recently, the prototype based research have been promoted into deep learning schemes, thus facilitating prototype learning in various aspects, including few-shot (Snell, Swersky, and Zemel 2017), zero (Jetley et al. 2015), unsupervised (Wu et al. 2018; He et al. 2020), and supervised (Yang et al. 2018) learning.

To the best of our knowledge, previous CSLR models rarely study the gloss representation via prototype learning. We observe the gloss-to-prototype correlation nature in gloss discrimination and leverage the superiority of prototype learning to facilitate the CSLR task.

Proposed Method: CSGC

We introduce a Cross-Sentence Consistency (CSGC) for CSLR. As illustrated in Fig. 2, our framework contains three main components: a gloss prototype (GP), a gloss contrastive loss function and an auxiliary similarity fusion strategy (ASFS).

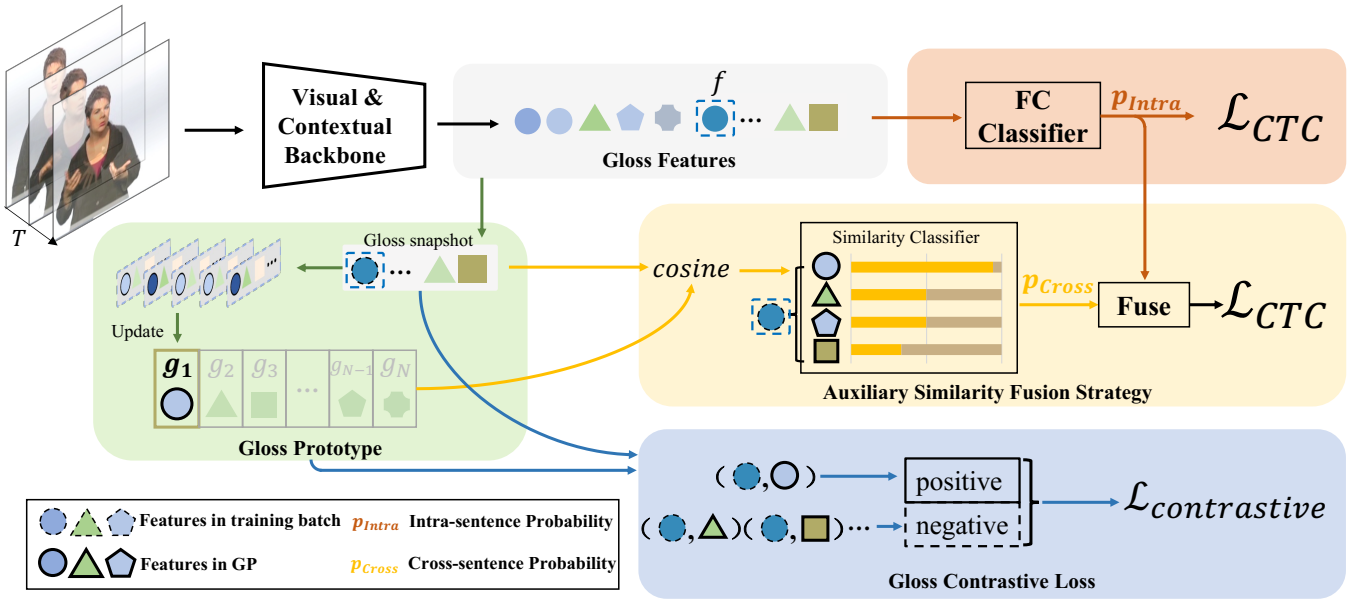

Figure 2: The CSGC framework is designed to analyze continuous video streams for sign language recognition. It first extracts gloss features and predictions from the video using a combination of visual-contextual backbones and an FC classifier. At its core, CSGC uses these predictions to operate its three main elements: (i) the Gloss Prototype, a comprehensive repository of gloss representations (left green box); (ii) a gloss contrastive loss that refines the feature learning between glosses and prototypes (right bottom box); and (iii) an auxiliary similarity fusion strategy that integrates cross-sentence gloss consistency to improve the accuracy of gloss recognition (right middle box).

Gloss Prototype

Gloss Prototype (GP) serves as a comprehensive feature dictionary within our framework, characterized by three primary operations: initialization, query (referral), and update. Functionally, it aggregates and maintains generalized feature representations, which are dilated from all gloss samples in memory across various categories, and undergoes continuous updating throughout the entire training process. We denote one segment of the gloss features (the output of the visual & contextual encoder) in a specific sample as $f \in \mathcal{R}^C$ (indicated by the blue dotted box in Fig. 2), where C is the channel size.

GP is initialized by an empirical distribution or a statistical distribution. It consistently maintains the representative gloss prototypes during the whole training, and contributes critical gloss representation references for following modules.

Referral with Memory As a global representation reference provider, GP can serve as a prototype memory bank. The query in GP can be formulated as a dictionary look-up task.

Since in CSLR task, there is no specific boundary label for gloss segments, we utilize the pseudo label produced by the CTC for gloss segments. Specifically, given the specific gloss feature f, a gloss representation reference $g_n \in G$ stored in GP is provided with expected category n, where the category index n is estimated by the intra-sentence learnable FC classifier (*i.e.*, p_{Intra} in Fig. 2). The referral of f is expressed as:

$$\begin{aligned} n &= \text{argmax}(p_{\text{Intra}}(f)), \\ \text{Ref}(f) &= g_n, \end{aligned} \tag{1}$$

where $p_{\text{Intra}}(f) = \text{Linear}_{C \to N}(f)$ is the learnable fully-connected classifier. C, N denote the channel size and the number of gloss categories, respectively.

Momentum Update To distill representative prototypes from diverse gloss samples, we propose to repressively refine the GP by a momentum update. Specifically, we update the referral n-th gloss prototype in GP by:

$$g_n^* = \beta f + (1 - \beta) g_n, \tag{2}$$

where $0 \leq \beta < 1$ is a scaling factor, $\text{Ref}(f) = g_n$ is the referral representation of f as described in Eq. 1. g_n^* is the updated representation after each iteration. The small β will encourage GP to focus on feature memory and obtain a stable statistic while the large β will encourage GP to focus on instance variances and obtain exquisite statistics.

A well-refined GP will provide representative gloss feature references and benefit the gloss contrastive learning.

Gloss Contrastive Loss Function Contrastive loss (Hadsell, Chopra, and LeCun 2006) is widely used to measure the distance between paired sample points based on their similarity. Here we utilize this form to constrain our gloss prototype learning: on the one hand closing the gloss-to-prototype distance between the gloss sample point and its belonging

prototype (as positive samples), on the other hand enlarging the gloss-to-prototype distances between the gloss sample point and other category prototypes (as negative samples), thus making both GP and gloss representations more distinguishable.

Given the set of GP $G = \{g_1, ..., g_N\}$ and the gloss feature f in a specific sample, we select the corresponding item with the matched label (*i.e.*, the pseudo label from the CTC) as a positive reference g_+, and the rest in G as negative references. Similar to (He et al. 2020), we consider the InfoNCE (Oord, Li, and Vinyals 2018), as a form of the contrastive loss function, in this paper:

$$\mathcal{L}_c = -\log \frac{\exp(f \cdot g_+)}{\sum_{n=1}^{N} \exp(f \cdot g_n)}. \tag{3}$$

With the help of GP, this contrastive learning constrain helps our model be more distinguishable in gloss feature representation (*i.e.*, less intra-class gloss variations and more inter-class gloss distances).

Auxiliary Similarity Fusion Strategy

To bring the benefits of the great representation capability from gloss prototype into predictions and achieve more comprehensive recognition clue distillation, we propose to solve the temporal classification in two aspects and fuse their results together: intra-sentence recognition clues and cross-sentence prototype-assisted recognition clues.

In spite of training a CTC classifier, *i.e.*, a fully connected layer that can handle gloss recognition within individual sentences (*i.e.*, p_{Intra} in Fig. 2), we propose an auxiliary similarity fusion strategy to tackle the gloss recognition in the aspect of gloss-to-prototype similarity by existing statistics of GP.

Prototype-Assisted Similarity Classifier. To exploit GP's great capability in representative modeling and bridge the gloss-to-prototype correlations, we propose an auxiliary similarity measurement. It is a pure statistic approach measuring the distance between a specific gloss sample with current GP state without learnable parameters. Specifically, cosine distances are calculated to estimate the similarity between a given sample feature and all items in GP:

$$D_f = \{d_{f,n} : 0 < n \leq N\}, \tag{4}$$

$$d_{f,n} = \cos(f, g_n) = \frac{f \cdot g_n}{||f||||g_n||}, \tag{5}$$

where $D_f \in \mathcal{R}^N$ containing the similarity information between the given gloss feature f and current snapshopt of GP.

We obtain the corss-sentence prototype-assisted similarity probability $p_{\text{Cross}}(f)$ by normalizing D_f with a softmax function:

$$p_{\text{Cross}}(f) = \{\frac{\exp(d_{f,n})}{\sum_{i=1}^{N} \exp(d_{f,i})} : 0 < i \leq N, 0 < n \leq N\}. \tag{6}$$

Intra-and-Cross Sentence Gloss Probability Fusion. The initial intra-sentence probability is obtained from a learnable FC classifier:

$$p_{\text{Intra}}(f) = \text{Linear}_{C \to N}(f), \tag{7}$$

where the linear function is implemented as a fully-connected layer. It is noted that the traditional FC classifier provide recognition clues within individual sentences, while our prototype-assited similarity classifier provide recognition clues across different sentences in a global (dataset) scope. Therefore, based on the two probabilities from the scope of intra-sentence and cross-sentence, we fuse them to obtain a more accurate probability as output. Regarding to the fusing way, we devise the following two feasible approaches:

(i) Sum. We fuse the two stream probabilities by summing:

$$p_o(f) = \text{Softmax}(p_{\text{Cross}}(f) + \alpha p_{\text{Intra}}(f)), \tag{8}$$

where α is a scaling factor.

(ii) Product. We fuse the two stream probabilities by production:

$$p_o(f) = \text{Softmax}(p_{\text{Cross}}(f) \cdot p_{\text{Intra}}(f)). \tag{9}$$

The output probability attends to the temporal classification and is supervised by a CTC (Graves et al. 2006) loss. We denote it as an auxiliary similarity fusion loss:

$$\mathcal{L}_f = \text{CTC}(p_o). \tag{10}$$

Objective Function

To supervise the training of our CSGC framework, we devise the objective function as a combination of four losses: a CTC loss (Graves et al. 2006) $\mathcal{L}_{\text{Seq}}$ for sequence classification, an visual enhancement loss $\mathcal{L}_{\text{VE}}$ (proposed in (Min et al. 2021)), and a gloss contrastive loss L_c (Eq. 3), an auxiliary similarity fusion loss L_f (Eq. 10). We combine these losses as our objective function:

$$\mathcal{L} = \mathcal{L}_{\text{Seq}} + \mathcal{L}_{\text{VE}} + \gamma_1 \mathcal{L}_c + \gamma_2 \mathcal{L}_f. \tag{11}$$

where γ_1 and γ_2 are both scaling factors. Since the scaling factor of $\mathcal{L}_{\text{Seq}}$ and $\mathcal{L}_{\text{VE}}$ is verified as $1:1$ in VAC (Min et al. 2021), we directly utilize this proportion in our experiments.

Experiments

Datasets and Metrics

We validate our proposed method on three datasets that are widely utilized in CSLR evaluation: PHOENIX14 (Koller, Forster, and Ney 2015), PHOENIX14-T (Camgoz et al. 2018) and CSL-Daily (Zhou et al. 2021).

PHOENIX14 (Koller, Forster, and Ney 2015) is a German CSLR dataset. This dataset contains representative sign videos collected from a weather TV program covering a wide range of signs and sentences. Specifically, its vocabulary covers 1295 signs. It provides 6841 video-sentence pairs in total. Following the official split (Koller, Forster, and Ney 2015), 5672, 540, 129 sentences are used for training, validation (Dev) and testing (Test), respectively.

PHOENIX14-T (Camgoz et al. 2018) is another German CSLR dataset, which is considered as an extension of (Koller, Forster, and Ney 2015). It has a vocabulary of 1085 signs. It contains 8247 video-sentence pairs in total,

which is divided into 7096 training instances, 519 validation instances (Dev) and 642 testing (Test) instances.

CSL-Daily (Zhou et al. 2021) is a large-scale Chinese CSLR dataset covering sign language scenarios in people's daily lives. Specifically, it has a vocabulary of 2000 signs and 20654 video-sentence pairs. The split of the dataset for training, validation (Dev) and testing (Test) is 18401, 1077 and 1176, respectively.

Word Error Rate (WER) is utilized as the metric to evaluate CSLR performance. It is defined as the minimal number of deletion, substitution and insertion operations on glosses when converting output sentences to ground-truth:

$$\mathrm{WER} = \frac{\#\mathrm{deletions} + \#\mathrm{substitutions} + \#\mathrm{insertions}}{\#\mathrm{glosses}}. \tag{12}$$

It can be inferred that the lower WER indicates better CSLR performance.

Moreover, we devise a Gloss Accuracy (GA) metric to evaluate the model performance on specific gloss categories. The details and related experiments are listed in the supplementary materials.

Implementation Details

Visual and contextual backbone. For fair comparison, we align the backbone settings with recent works (Niu and Mak 2020; Min et al. 2021; Hao, Min, and Chen 2021; Hu et al. 2022). Specifically, we use ResNet18 (He et al. 2016) as visual backbone. Except for the visual inputs, we do not use other modality clues for simplicity. The short-term temporal convolution module contains a 5-kernel size convolution, a 2-kernel size max pooling and another 5-kernel size convolution, sequentially, *i.e.*, K5-P2-K5. In long-term sequence modeling, we utilize a two-layer BiLSTM with 1024 hidden states.

Designs of Gloss Prototype. GP stores prototypical features of all gloss categories. It can be initialized by empirical distributions or a statistical distribution. We denote it as $G = \{g_n : 0 < n \leq N\}$, where g_n represents the prototypical feature of n-th gloss, N denotes the number of gloss categories, $g_n \in \mathcal{R}^C$. We devise two empirical distributions as the initialization, *i.e.*, 1) zero initialization: $G = \{g_n = 0 : 0 < n \leq N\}$ and 2) Gaussian initialization: $G \sim \mathcal{N}(\mu, \sigma^2)$, where parameters μ and σ^2 are the mean and variance. We empirically set $\mu = 0, \sigma = 1$ in our experiments.

Another initialization approach is scratching from statistics. Suppose a basic CSLR model can provide a rough reference of how gloss features distribute, thus we can initialize the GP by historical statistics. Specifically, we evaluate the basic model on the entire dataset and average all feature representations as gloss feature initialization, grouped by their predicted gloss categories:

$$\begin{aligned} G &= \{g_n : 0 < n \leq N\}, \\ g_n &= \frac{\sum_{v \in V} \sum_{p^t = n}^{m_v} f^t}{\sum_{v \in V} m_v}, \\ \{(p^t, f^t) : 0 < t \leq T\} &= \mathcal{B}_{\mathrm{CNN}}(v), \end{aligned} \tag{13}$$

Model	Prototype Scope	WER (%) Dev	WER (%) Test
baseline-VAC	w/o prototype	21.2	22.3
baseline-clean	w/o prototype	20.7	21.1
baseline-clean	1-sentence	20.0	20.5
	4-sentence	20.0	20.7
	16-sentence	19.7	20.5
baseline-clean	global (GP)	**19.2**	**19.7**

Table 1: Performance with different scopes of GP.

Model	Initialization	WER (%) Dev	WER (%) Test
baseline-clean	-	20.7	21.1
GP	zeros	19.7	20.4
	random	19.6	20.1
	statistics	**19.2**	**19.7**

Table 2: Performance with different GP initializations.

where v denotes the input video, $\mathcal{B}_{\mathrm{CNN}}$ denotes the basic CNN model, p^t and f^t are prediction and corresponding gloss feature along the time sequence. m_v denotes the number of matched samples in a video v. V specifies the whole video dataset. Experiments in Table 2, Sec. indicate our GP is not very sensitive to the initialization approach, despite the statistic initialization performs slightly better.

GP is updated periodically during the training. To obtain statistical representative features on the entire dataset, a direct way is summing up all gloss features with the same prediction and then averaging them. However, this approach yields performance degradation (as illustrated in Table 3, Sec.). We reckon it is due to the fact that less-discriminated features in early stages count the same as well-discriminated features in late stages during the whole training, thus producing less representative statistics. To alleviate this phenomenon, we propose to repressively refine the GP by the momentum update (Sec.).

To ensure the reproducibility of our work, detailed training and inference settings are included in the supplementary materials.

Ablation Studies

To demonstrate the effectiveness of our CSGC framework, we conduct ablation studies on separate model components. For fair comparison, all experiments are conducted on the PHOENIX14 dataset using a unified backbone ResNet18. We utilize an existing work VAC (Min et al. 2021) as a strong baseline.

An optimized strong baseline. On the basis of the original version (denoted as baseline-VAC), we make two modifications to make the baseline clean and simpler. Firstly, we remove its alignment constraint thus only keeping the visual enhancement constraint. Secondly, we remove the last pooling layer in its temporal convolution module (*i.e.*, from K5-

Method	Avg.	Momentum Update (β) 0.3	0.5	0.7	0.9	0.99
WER (%)	21.1	20.8	20.5	19.8	**19.2**	19.7

Table 3: An ablation study of momentum update. (Dev set)

Model	WER (%) Dev	Test
GP (+)	19.2	19.7
GP (+&-)	**18.5**	**19.4**

Table 4: Performance (WER) comparison between a pure consistency learning scheme (+) and a gloss contrastive learning scheme (+&-).

Component FC	GP	AS	WER (%) Dev	Test
✓	✗	✗	20.7	21.1
✓	✓	✗	18.5	19.4
✓	✓	✓	**18.1**	**19.0**

Table 5: Performance (WER) comparison among different component combinations. FC denotes the fully-connected classifier, AS denotes the auxiliary similarity classifier.

Strategy	Sum (α) 0.2	0.5	1.0	2.0	5.0	Product
WER (%)	20.4	20.1	19.7	18.9	18.8	**18.1**

Table 6: Performance (WER) comparison between different fusion strategies (sum vs. product) on Dev set.

P2-K5-P2 to K5-P2-K5). We denote this modified version baseline as baseline-clean.

Baseline-clean achieves better performance (*i.e.*, Dev 20.7, Test 21.1 on PHOENIX14 dataset) than the original baseline-VAC. Detailed ablations and analysis are listed in supplementary materials.

Effectiveness of the GP. GP is proposed to improve the gloss feature discrimination via cross-sentence gloss-to-prototype learning. The prototype is maintained within a specific updating scope and supervised by a gloss contrastive loss. To verify its effectiveness, we analyze the intuition behind the GP by extending the prototype at different scopes: without any prototype, prototype within limited sentences, the global level (entire dataset) prototype (*i.e.*, GP). All experiments are conducted using a unified pipeline (*i.e.*, backbone, TC, BiLSTM and CTC loss). As illustrated in Table 1, comparing with a simple baseline without prototype, models with sentence scope prototype demonstrate performance improvements. When equipped with global scope prototype (GP), as the last row illustrates, further performance improvement is observed.

Impacts of GP initializations. Due to the statistical nature of the prototype, the quality of initialization is critical. We study the impacts of different approaches to initialize GP, as shown in Table 2. We devise three approaches to initialize GP. The first approach is to initialize each category in GP by a zero vector (GP-zeros). The second approach is to randomly initialize each category vector in GP (GP-random). The last approach is to initialize each category vector in GP by statistics, *i.e.*, we utilize an off-the-shelf baseline model (baseline-clean in Table 1), to calculate average feature representations over the entire training set and set them as initialization for all gloss categories. As Table 2 indicates, initializing from statistics achieves better performance, therefore we utilize this approach as default initialization of GP in following experiments.

Impacts of Momentum update. We ablate the effectiveness of momentum update in two aspects: (i) comparing the momentum update approach with an intuitive global averaging approach and (ii) the impacts of different momentum values (*i.e.*, β in Eq. 2). As illustrated in Table 3, comparing with an intuitive global average approach, the momentum update approach demonstrates better performance. Particularly, the momentum update approach performs reasonably well with a proper momentum value (*i.e.*, $\beta = 0.9$).

Effectiveness of the contrastive loss. The effectiveness of GP is verified aforementioned. The performance in Table 1-2 is limited because we utilize a pure similarity learning scheme without negative samples. For better gloss representation learning, we further improve the scheme by introducing a contrastive loss. This gloss contrastive loss not only enhance the consistency among glosses belonging to the same prototype (positive samples), but also enlarge the distances with different category prototypes (negative samples). We compare the pure GP based similarity learning approach (only positive samples) with the contrastive loss version (positive and negative samples). As Table 4 illustrates, a performance gain is observed when utilizing the contrastive loss.

Effectiveness of the proposed auxiliary similarity fusion strategy. There are two branches to estimate gloss probabilities in our framework, *i.e.*, an intra-sentence FC classifier (noted as FC, producing p_{Intra}) and a cross-sentence auxiliary similarity classifier (noted as AS, producing p_{Cross}). ASFS fuses their recognition probabilities. Table 5 shows the performance variations among different branch combinations. Specifically, row 2 indicates the overall effectiveness of GP whose inner designs (*i.e.*, initialization, momentum update and contrastive loss) are ablated aforementioned. The performance gain of fusing the two probabilities (row 3) verifies that our proposed auxiliary similarity fusion strategy has a positive influence.

Impacts of fusion factors. Table 6 ablates different fusion factors when combining the two branch probabilities together. Different scaling factors (α in Eq. 8) are ablated when utilizing the sum strategy. We note that product strategy demonstrates better effect.

Impacts of loss weights. There are four losses in our model, *i.e.*, a visual-enhancement CTC, a sequence CTC, the proposed similarity loss and a contrastive loss. Each loss is

Model	Backbone	PHOENIX14		PHOENIX14-T		CSL-Daily	
		Dev	Test	Dev	Test	Dev	Test
SubUNets (Cihan Camgoz et al. 2017)	CaffeNet	40.8	40.7	-	-	41.4	41.0
Staged-Opt (Cui, Liu, and Zhang 2017)	VGG-S	39.4	38.7	-	-	-	-
Align-iOpt (Pu, Zhou, and Li 2019)	3D-ResNet	37.1	36.7	-	-	-	-
SFL (Niu and Mak 2020)	ResNet18	26.2	26.8	25.1	26.1	-	-
C+L+H (Koller et al. 2019)*	GoogLeNet	26.0	26.0	22.1	24.1	-	-
STMC (Zhou et al. 2020)*	VGG11	21.1	20.7	19.6	21.0	-	-
DNF (Cui, Liu, and Zhang 2019)*	GoogLeNet	23.8	24.4	-	-	32.8	32.4
FCN (Cheng et al. 2020)	Custom	23.7	23.9	23.3	25.1	33.2	32.5
CMA (Pu et al. 2020)	GoogLeNet	21.3	21.9	-	-	-	-
VAC (Min et al. 2021)	ResNet18	21.2	22.3	-	-	-	-
SMKD (Hao, Min, and Chen 2021)	ResNet18	20.8	21.0	20.8	22.4	-	-
C^2SLR (Zuo and Mak 2022)	VGG11	20.5	20.4	-	-	-	-
TLP (Hu et al. 2022)	ResNet18	19.7	20.8	-	-	-	-
Joint-SLRT (Camgoz et al. 2020)	Inception	-	-	24.6	24.5	33.1	32.0
SLT (Camgoz et al. 2018)*	GoogLeNet	-	-	24.5	24.6	-	-
LS-HAN (Huang et al. 2018)	Custom	-	-	-	-	39.0	39.4
SignBT (Zhou et al. 2021)	Transformer	-	-	-	-	33.6	33.1
Ours	ResNet18	**18.1**	**19.0**	**17.2**	**19.5**	**27.1**	**26.7**

Table 7: Quantitative evaluations on PHOENIX14 (Koller, Forster, and Ney 2015) , PHOENIX14-T (Camgoz et al. 2018) and CSL-Daily (Zhou et al. 2021) datasets. Results are quantified by percentages (%). * indicates using extra clues (i.e., hand, mouth or face gestures).

assigned a weight factor during optimization. We study the impacts of the weights in supplementary materials. The results indicate that as loss proportion varies, the performance of our CSGC model is degraded in less than 1%, which implies the robustness of our gloss contrastive loss and the fusion strategy. We empirically adopt the same setting in our experiments, *i.e.*, $\gamma_1 = 0.3, \gamma_2 = 0.1$.

Comparison with State-of-the-arts

We compare our result with existing state-of-the-arts on three benchmarks: PHOENIX14 (Koller, Forster, and Ney 2015), PHOENIX14-T (Camgoz et al. 2018) and CSL-Daily (Zhou et al. 2021). We select a representative state-of-the-art method VAC (Min et al. 2021) and make detailed comparisons with it. We devise a Gloss Accuracy (GA) metric to evaluate the performance on concrete gloss categories. GA is calculated as a ratio of the correct gloss number to the total gloss number. The GA differences on a part of gloss categories are represented in the supplementary materials. A meanGA comparison illustrated in supplementary materials indicating a large 10.8% accuracy improvement on VAC result. These remarkable improvements on not only the particular gloss categories but also the overall mean accuracy indicate the effectiveness of our CSGC on gloss discrimination. Moreover, we visualize the center point t-SNE distributions of our CSGC model (*i.e.*, vectors in GP) and VAC model (*i.e.*, averaged features grouped by gloss categories) in Fig. 4 in supplementary materials. This comparison indicates more discriminative distributions of our GP regarding the distributions of center points of VAC model, thanks to the critical cross-sentence discrimination learned from our CSGC.

We make performance comparisons with all existing CSLR works in Table 7, by the broadly used metric WER. We notice that even though some works use additional information for better performance, such as face or hand gestures (as * indicates), our method still surpasses all existing works by a big margin, only using visual information. Remarkably, we improve current state-of-the-art performance on the Dev sets of PHOENIX14 (Koller, Forster, and Ney 2015), PHOENIX14-T (Camgoz et al. 2018), and CSL-Daily (Zhou et al. 2021) by **1.6**%, **2.4**%, and **5.7**% respectively, and by **1.4**%, **1.5**%, **5.3**% on their Test sets respectively. The significant improvement to the state-of-the-art demonstrates the effectiveness of our method.

Conclusion

In this paper, we present a cross-sentence gloss consistency for CSLR. We first observe the limitation of current representation consistency based learning within individual sentences and present a gloss prototype, aiming at cross-sentence gloss discrimination learning. Benefiting from our well-distinguished gloss prototypes, our model significantly improves the gloss discrimination with a gloss contrastive loss and an auxiliary similarity fusion strategy, thus achieving better gloss recognition. Extensive experiments verify the effectiveness of our proposed framework. Remarkably, our framework extends the performance boundary on the existing three benchmarks by 1.6%, 2.4% and 5.7% large margins. We believe that our cross-sentence gloss consistency will bring a flurry innovation to the research field and profoundly contribute to the development of CSLR.

References

Aamodt, A.; and Plaza, E. 1994. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1): 39–59.

Camgoz, N. C.; Hadfield, S.; Koller, O.; Ney, H.; and Bowden, R. 2018. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7784–7793.

Camgoz, N. C.; Koller, O.; Hadfield, S.; and Bowden, R. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10023–10033.

Cheng, K. L.; Yang, Z.; Chen, Q.; and Tai, Y.-W. 2020. Fully convolutional networks for continuous sign language recognition. In *European Conference on Computer Vision*, 697–714. Springer.

Cihan Camgoz, N.; Hadfield, S.; Koller, O.; and Bowden, R. 2017. Subunets: End-to-end hand shape and continuous sign language recognition. In *Proceedings of the IEEE international conference on computer vision*, 3056–3065.

Cui, R.; Liu, H.; and Zhang, C. 2017. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7361–7369.

Cui, R.; Liu, H.; and Zhang, C. 2019. A deep neural framework for continuous sign language recognition by iterative training. *IEEE Transactions on Multimedia*, 21(7): 1880–1891.

Duda, R. O.; Hart, P. E.; and Stork, D. G. 1973. *Pattern classification and scene analysis*, volume 3. Wiley New York.

Freeman, W. T.; and Roth, M. 1995. Orientation histograms for hand gesture recognition. In *International workshop on automatic face and gesture recognition*, volume 12, 296–301. Zurich, Switzerland.

Gao, W.; Fang, G.; Zhao, D.; and Chen, Y. 2004. A Chinese sign language recognition system based on SOFM/SRN/HMM. *Pattern Recognition*, 37(12): 2389–2402.

Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, 369–376.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, 1735–1742. IEEE.

Han, J.; Awad, G.; and Sutherland, A. 2009. Modelling and segmenting subunits for sign language recognition based on hand motion analysis. *Pattern Recognition Letters*, 30(6): 623–633.

Hao, A.; Min, Y.; and Chen, X. 2021. Self-Mutual Distillation Learning for Continuous Sign Language Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11303–11312.

Hastie, T.; Tibshirani, R.; Friedman, J. H.; and Friedman, J. H. 2009. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hu, L.; Gao, L.; Liu, Z.; and Feng, W. 2022. Temporal lift pooling for continuous sign language recognition. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV*, 511–527. Springer.

Huang, J.; Zhou, W.; Zhang, Q.; Li, H.; and Li, W. 2018. Video-based sign language recognition without temporal segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Jetley, S.; Romera-Paredes, B.; Jayasumana, S.; and Torr, P. 2015. Prototypical priors: From improving classification to zero-shot learning. *arXiv preprint arXiv:1512.01192*.

Koller, O.; Camgoz, N. C.; Ney, H.; and Bowden, R. 2019. Weakly supervised learning with multi-stream CNN-LSTM-HMMs to discover sequential parallelism in sign language videos. *IEEE transactions on pattern analysis and machine intelligence*, 42(9): 2306–2320.

Koller, O.; Forster, J.; and Ney, H. 2015. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141: 108–125.

Koller, O.; Zargaran, O.; Ney, H.; and Bowden, R. 2016. Deep sign: Hybrid CNN-HMM for continuous sign language recognition. In *Proceedings of the British Machine Vision Conference 2016*.

Koller, O.; Zargaran, S.; and Ney, H. 2017. 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*, 4297–4305.

Min, Y.; Hao, A.; Chai, X.; and Chen, X. 2021. Visual alignment constraint for continuous sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11542–11551.

Newell, A.; Simon, H. A.; et al. 1972. *Human problem solving*, volume 104. Prentice-hall Englewood Cliffs, NJ.

Niu, Z.; and Mak, B. 2020. Stochastic fine-grained labeling of multi-state sign glosses for continuous sign language recognition. In *European Conference on Computer Vision*, 172–186. Springer.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Pu, J.; Zhou, W.; Hu, H.; and Li, H. 2020. Boosting continuous sign language recognition via cross modality augmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, 1497–1505.

Pu, J.; Zhou, W.; and Li, H. 2019. Iterative alignment network for continuous sign language recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4165–4174.

Shawe-Taylor, J.; Cristianini, N.; et al. 2004. *Kernel methods for pattern analysis*. Cambridge university press.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wu, Z.; Xiong, Y.; Yu, S. X.; and Lin, D. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3733–3742.

Yang, H.-M.; Zhang, X.-Y.; Yin, F.; and Liu, C.-L. 2018. Robust classification with convolutional prototype learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3474–3482.

Yang, Y.; Zhuang, Y.; and Pan, Y. 2021. Multiple knowledge representation for big data artificial intelligence: framework, applications, and case studies. *Frontiers of Information Technology & Electronic Engineering*, 22(12): 1551–1558.

Zhou, H.; Zhou, W.; Qi, W.; Pu, J.; and Li, H. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1316–1325.

Zhou, H.; Zhou, W.; Zhou, Y.; and Li, H. 2020. Spatial-temporal multi-cue network for continuous sign language recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 13009–13016.

Zuo, R.; and Mak, B. 2022. C2SLR: Consistency-Enhanced Continuous Sign Language Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5131–5140.