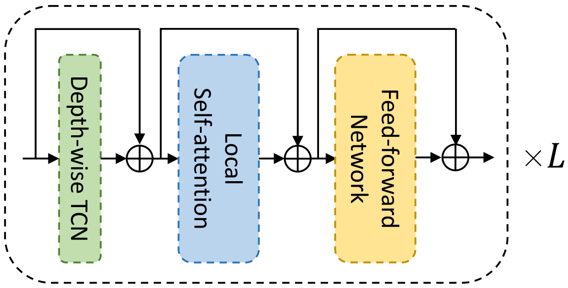
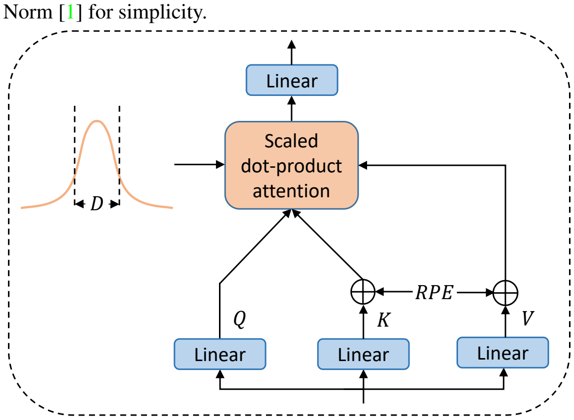
A. Supplementary Materials

A.1. Local Transformer



(a) The architecture of Local Transformer (LT). We omit Layer-Norm [1] for simplicity.



(b) The architecture of Local Self-attention (LSA).

Figure 8. We propose a strong sequential module, *i.e.*, Local Transformer (LT), which is used in the VLT backbone. It is built based on QANet [14], which validates the effectiveness of combining TCNs with self-attention, and the difference is that we further leverage Gaussian bias [5, 13] to introduce local contexts to the self-attention module, *i.e.*, Local Self-attention (LSA). (L: the number of LT layers, we set it to 2 as default; RPE: relative positional encoding [10]; D: the window size of the Gaussian bias.)

Sequence modeling plays a key role in the CSLR task. Capturing long-term temporal dependencies was proven to be effective on many sequence modeling tasks, *e.g.*, neural machine translation [12], and speech recognition [4]. Thus, it is reasonable to introduce globally-guided architectures, *e.g.*, BiLSTM [8, 9] and vanilla Transformer [2, 7], to the CSLR task. However, within a sign language video, each gloss is short, consisting of only a few frames. This can be the reason why a locally-guided architecture, *i.e.*, TCNs, has also been adopted to CSLR successfully [3]. Motivated by this, we propose a mixed architecture, Local Transformer (LT), to leverage both global and local contexts for sequence modeling for CSLR.

As shown in Figure 8a, each LT layer consists of a depth-wise TCN layer, a local self-attention (LSA) layer, and a feed-forward network. Since the depth-wise TCN layer and the feed-forward network are the same as those used in [12, 14], below we only formulate the LSA layer.

As shown in Figure 8b, given a feature sequence $\mathbf{Z} \in \mathbb{R}^{T\times d}$, three separate linear layers first project $\mathbf{Z}$ into queries $\mathbf{Q} \in \mathbb{R}^{T\times d}$, keys $\mathbf{K} \in \mathbb{R}^{T\times d}$, and values $\mathbf{V} \in \mathbb{R}^{T\times d}$, respectively. We adopt multi-head self-attention which is more effective than its single-head counterpart [12] by splitting $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ into $\{\mathbf{Q}^h\}_{h=1}^{N_h}, \{\mathbf{K}^h\}_{h=1}^{N_h}, \{\mathbf{V}^h\}_{h=1}^{N_h}$, respectively, where $\mathbf{Q}^h, \mathbf{K}^h, \mathbf{V}^h \in \mathbb{R}^{T\times d/N_h}$ and N_h is the number of heads. Then scaled dot-product attention [5, 12] is used to compute the attention scores for each head as follows:

$$\mathbf{scores} = \left\{ \frac{(\mathbf{Q}^h)(\mathbf{K}^h)'}{\sqrt{d/N_h}} \right\}_{h=1}^{N_h} \in \mathbb{R}^{N_h\times T\times T}. \tag{16}$$

In order to model local contexts, we adopt Gaussian bias to emphasize the relations between close query-key (QK) pairs and weaken the relations between distant QK pairs. Given a QK pair $(\mathbf{q}_i^h, \mathbf{k}_j^h)$, the Gaussian bias is defined as:

$$bias_{ij}^h = -\frac{(j-i)^2}{2\sigma^2}, \tag{17}$$

where $\sigma = \frac{D}{2}$, and D is the window size of the Gaussian bias [5]. The Gaussian bias is *head-shared*; that is, it is common among the heads since Eq. 17 is independent to h. Then the attention weights of each value vector are obtained from a softmax layer, and the output of the self-attention module is:

$$\begin{cases} \mathbf{O}^h = softmax(\mathbf{scores}^h + \mathbf{bias}^h)\mathbf{V}^h \\ \mathbf{O}^{SA} = concat(\{\mathbf{O}^h\}_{h=1}^{N_h})\mathbf{W}^O \in \mathbb{R}^{T\times d} \end{cases}, \tag{18}$$

where $\mathbf{W}^O \in \mathbb{R}^{d\times d}$ denotes the output linear layer.

In terms of the choice of D, we consider that the ratio of frame length to gloss sequence length, *i.e.*, T_i/N_i, where i denotes i-th training sample, is a good estimate of the window size as it represents the average frame length of a gloss, which is similar to the idea of the window size. Thus, we set D as:

$$D = \frac{1}{|tr|}\sum_{i=1}^{|tr|}\frac{T_i}{N_i}, \tag{19}$$

where $|tr|$ is the number of training samples. More specifically, $D = 6.3, 6.3, 15.8$ for the PHOENIX-2014, PHOENIX-2014-T, and CSL dataset, respectively.

We conduct ablation studies to validate the effectiveness of the LSA and the depth-wise TCN (DTCN) layer. As shown in Table 9, both the LSA and the DTCN can clearly improve the model's performance, which establishes our LT as a strong sequential module for the CSLR task.

Method	LSA	DTCN	WER%
VGG11+TF	×	×	25.2
	✓	×	22.7
	✓	✓	**21.5**

Table 9. Ablation study for the local Transformer. (TF: Transformer; LSA: local self-attention; DTCN: depth-wise TCN.)

Factor	1	5	10	15	20	25
Dev	**22.3**	**22.3**	22.8	23.1	23.2	22.6
Test	23.4	**22.8**	22.9	**22.8**	23.7	23.4

Table 10. Fine-tuning results of VAC [6] on the VLT backbone.

A.2. Fine-tuning Results of VAC

We compare VAC with our SEC as shown in Table 4 in the main section. For fair comparisons, we fine-tuned the factor of the VA loss as [6] on the VLT backbone based on the open-sourced codes[1]. As shown in Table 10, the optimal factor is 5.

A.3. Choice of γ_x, γ_y

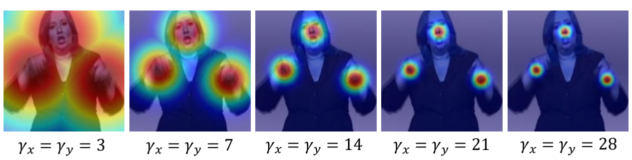


Figure 9. Visualization results for different γ_x, γ_y. Since for real practice, the height and the width of the spatial attention masks are usually the same, we set γ_x and γ_y to the same value.

γ_x, γ_y	3	7	14	21	28
Dev	21.3	21.2	**21.1**	21.4	21.3
Test	21.7	21.9	**20.8**	21.5	21.6

Table 11. Comparison among different γ_x, γ_y

We conduct experiments to compare the performance of different γ_x, γ_y as shown in Table 11. Among them, either too large γ_x, γ_y (cannot cover entire informative regions) or too small γ_x, γ_y (cover too many trivial regions) can harm the model's performance. The model can achieve the best performance when $\gamma_x = \gamma_y = 14$.

A.4. Using Keypoints Heatmaps as Filters

In this work, we use keypoints heatmaps as guidance for the spatial attention module. To better validate its effectiveness, we conduct one more experiment that directly use keypoints heatmaps as filters to modulate the feature maps, *i.e.*, multiplying the feature maps with the keypoints heatmaps directly. However, as shown in Table 12, using heatmaps as filters can damage the model's performance. We think this is because when we use heatmaps as guidance, the visual module can be enforced to concentrate on informative regions by $\mathcal{L}_{SAC}$, but this enforcement is absent if we simply use keypoints as filters.

[1] https://github.com/ycmin95/VAC_CSLR

Method	WER%
Filters	22.9
Guidance	**20.8**

Table 12. Comparison between using keypoints heatmaps as filters and guidance for the spatial attention module.

A.5. Visualization Results for $\mathcal{L}_{SEC}$.

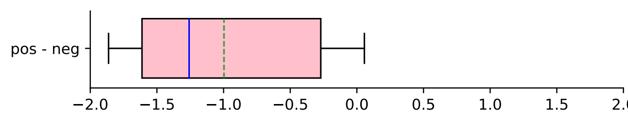


Figure 10. The box plot of the difference between positive and negative distance in $\mathcal{L}_{SEC}$. The blue line denotes the median, and the green line denotes the mean.

We draw a box plot on the difference between the positive and negative distance in $\mathcal{L}_{SEC}$ as shown in Figure 10. Since our distance function $d(\cdot,\cdot) = 1 - cos(\cdot,\cdot)$, the differences must lie in $[-2, 2]$. According to the position of the median, almost half of the batches can achieve a large difference (≤ -1.25). Also, most of the batches (at least 75%) have a smaller positive distance (difference < 0). This means that the positive and negative samples are well-separated, which again validates the effectiveness of our SEC.

A.6. Discussion

As shown in Table 2 in the main section, the effectiveness of the post-processing module implies that the quality of the pose keypoints heatmaps plays a key role in our SAC. Although our post-processing module can refine the original heatmaps, the refined heatmaps may not be optimal. In the future, we will try to co-train the keypoints heatmap extractor, *e.g.*, HRNet [11], with the CSLR backbones to yield better heatmaps. However, the co-training must introduce more parameters and cost more GPU memory, thus there is a trade-off between the co-training and our method.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 1

[2] Necati Cihan Camgöz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *CVPR*, pages 10020–10030, 2020. 1

[3] Ka Leong Cheng, Zhaoyang Yang, Qifeng Chen, and Yu-Wing Tai. Fully convolutional networks for continuous sign language recognition. In *ECCV*, volume 12369, pages 697–714, 2020. 1

[4] Linhao Dong, Shuang Xu, and Bo Xu. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. In *ICASSP*, pages 5884–5888, 2018. 1

[5] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421, 2015. 1

[6] Yuecong Min, Aiming Hao, Xiujuan Chai, and Xilin Chen. Visual alignment constraint for continuous sign language recognition. In *ICCV*, pages 11542–11551, October 2021. 2

[7] Zhe Niu and Brian Mak. Stochastic fine-grained labeling of multi-state sign glosses for continuous sign language recognition. In *ECCV*, pages 172–186, 2020. 1

[8] Junfu Pu, Wengang Zhou, Hezhen Hu, and Houqiang Li. Boosting continuous sign language recognition via cross modality augmentation. In *ACM MM*, pages 1497–1505, 2020. 1

[9] Junfu Pu, Wengang Zhou, and Houqiang Li. Iterative alignment network for continuous sign language recognition. In *CVPR*, pages 4165–4174, 2019. 1

[10] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *NAACL-HLT*, pages 464–468, 2018. 1

[11] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pages 5693–5703, 2019. 2

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 1

[13] Baosong Yang, Zhaopeng Tu, Derek F. Wong, Fandong Meng, Lidia S. Chao, and Tong Zhang. Modeling localness for self-attention networks. In *EMNLP*, pages 4449–4458, 2018. 1

[14] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. Qanet: Combining local convolution with global self-attention for reading comprehension. In *ICLR*, 2018. 1