Action-Concentrated Embedding Framework: This is your captain sign-tokening

Hyunwook Yu[1*], Suhyeon Shin[2*], Jungu Heo[3], Hyuntaek Shin[4], Hyosu Kim[5†], Mucheol Kim[6†]
Computer Science and Engineering, Chung-Ang University[1,2,5,6]
UNIVIA Inc.[3,4]
{[1]yu990410, [2]girinssh, [5]hskimhello, [6]kimm}@cau.ac.kr,
[3]heojunku@univia.co.kr,[4]dorim123@gmail.com

Abstract

Sign language is the primary communication medium for people who are deaf or have hearing loss. However, given the divergent range of sensory abilities of these individuals, there is a communication gap that needs to be addressed. In this paper, we present action-concentrated embedding (ACE), which is a novel sign token embedding framework. Additionally, to provide a more structured foundation for sign language analysis, we introduce a dedicated notation system tailored for sign language that endeavors to encapsulate the nuanced gestures and movements that are integral with sign communication. The proposed ACE approach tracks a signer's actions based on human posture estimation. Tokenizing these actions and capturing the token embedding using a short-time Fourier transform encapsulates the time-based behavioral changes. Hence, ACE offers input embedding to translate sign language into natural language sentences. When tested against a disaster sign language dataset using automated machine translation measures, ACE notably surpasses prior research in terms of translation capabilities, improving the performance by up to 5.79% for BLEU-4 and 5.46% for ROUGE-L metric.

Keywords: sign language translation, sign language token embedding framework, short-time Fourier transform

1. Introduction

The transformer (Vaswani et al., 2017) paves a new way for communicating between intercultural societies (Araabi and Monz, 2020; Chi et al., 2021). However, its benefits are limited to spoken languages, meaning people who are deaf or have hearing loss remain on the sidelines of communication. As a result, an information imbalance persists within the deaf community (Lillo-Martin et al., 2023). To overcome this communication barrier, neural machine translation with sign language is gaining attention.

An understanding of the deep structures of the source language is essential to fully convey the original meaning, resulting in more accurate and complete translation results (Dai et al., 2022). This also applies to sign language, which is a unique linguistic system that has its own grammar, phonology, and lexicon.

Sign language translation approaches that simply represent a sequence of consecutive images overlook the intrinsic linguistic nature of sign languages (Camgoz et al., 2020; Yin and Read, 2020; Angelova et al., 2022). In other words, they lack sufficient consideration of sign language as a complex linguistic system in which each action represents stems and affixes (Müller et al., 2023). This ambiguity in the definition of tokens creates difficulties in terms of the consistent interpretation of linguistic elements and increases the time complexity of transformer models, which is proportional to the square of the number of tokens (Vaswani et al., 2017). To convey the essence of sign language, the machine translation of sign language needs to be rethought. Many natural language processing approaches analyze the morphological structure of languages. Furthermore, the phonology system of sign language is similar to that of spoken languages, indicating that sign communication should be treated as a language (Yin et al., 2021; Angelova et al., 2022)

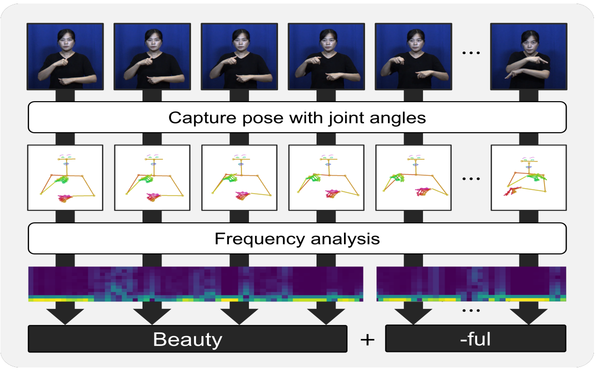

Figure 1: Concept of notation for sign language with a set of joints of posture

In speech processing using frequency analysis, short-time Fourier transform (STFT) is a key

[*] Equal Contribution
[†] Co-Corresponding Authors

LREC-COLING 2024, pages 310–320

method for representing the pronunciation of words as frequency characteristics over time and recognizing syllables in human speech (Mohd Hanifa et al., 2021). Sign language relies on frequency analysis to effectively capture gestures, as in spoken languages, where speech content and speaker identification are available (Beaudry et al., 2014). Frequency domain data can represent information from multiple frames in a high-dimensional form. Moreover, it can increase the efficiency of data processing and significantly reduce the time complexity of transformer models (Griffin and Lim, 1984).

Analyzing the frequency characteristics of sign language over time offers a new methodology in sign language translation. The signals sampled from actions (similar to voice analysis) are represented as frequency characteristics over time. This indicates that frequency analysis can distinguish given actions (Tran et al., 2014). Ultimately, tokenizing sign language is analogous to how we recognize words by listening to sounds. Hence, angles within a "pose" can be defined as an alphabet, and successive angle changes within a portion of a frame sequence can be interpreted as stems or affixes. The Figure 1 explains the concept of notation for sign language with a set of joints of posture.

In this paper, we propose the action-concentrated embedding (ACE) framework, which focuses on changes in the speaker's hands, face, and body motions. The ACE framework captures the posture changes of body parts over time and provides token embeddings using the STFT. In addition, it understands each action in the sign as a linguistic system of stems and affixes. The proposed approach effectively represents various actions of sign language and facilitates the understanding of sign language based on patterns of part-specific movement changes. Finally, the proposed framework improves sign language translation using a transformer-based model.

The remainder of the paper is structured as follows: Section 2 presents a review of related work in sign language recognition and translation, highlighting the gaps that the ACE framework addresses. Section 3 details the ACE framework, from angular data extraction to token embedding. Experiments validating ACE, including results and comparisons, are presented in Section 4. Section 5 presents the conclusions, a summary of the contributions, and suggestions for future research directions in sign language translation.

2. Related Work

2.1. Sign Language Recognition

Sign language recognition (SLR) is a branch of computational sign language that recognizes gestures to understand their meaning. Several studies have conducted research on recognizing human signatures, including sensing- and vision-based approaches.

The sensing-based approach is representative of the way in which wearable devices are manufactured (Wen et al., 2021; Zhou et al., 2020b; Zhang et al., 2023a), with non-contact approaches including radar (Rahman et al., 2021) and radio frequency (Ma et al., 2018; Zhang et al., 2020). However, these technologies have the disadvantage of requiring additional equipment.

In vision-based approaches, 3DCNN is the key method used for feature extraction (Huang et al., 2015; Al-Hammadi et al., 2020a,b). In addition to these traditional CNN-based approaches, pose-based methods that can incorporate human information have also been developed. These can extract human key points using pose estimation frameworks, such as OpenPose (Cao et al., 2017; Simon et al., 2017; Wei et al., 2016) or MediaPipe (Lugaresi et al., 2019). The work of Jiao et al. (2023); de Amorim et al. (2019); Tunga et al. (2021) are typical approaches that are based on the graph convolution network (GCN). Ko et al. (2018); Amaliya et al. (2021) recognized sign language by the key point on the image, where word-level SLR determines a gloss from a piece of data. Because continuous SLR (CSLR) recognizes gloss in continuous sign language data, temporal information becomes very important. Many researchers have attempted to extract temporal features with different models, including the hidden Markov model (Koller et al., 2017, 2016), long short-term memory (Yang and Zhu, 2017; Basnin et al., 2020) and both (Koller et al., 2019). Recently, CNN-based models, such as spatio-temporal multi-cue network (Zhou et al., 2020a) and 3D-CNN (Zhang et al., 2023b), have also been proposed.

2.2. Sign Language Translation

Although CSLR conveys sign language meaning, it is difficult for hearing people to fully understand sentences expressed through it. The Neural Sign Language Translation (NSLT) paper formally proposes a deep learning-based SLT (Camgoz et al., 2020)), which includes sign-to-text (S2T), gloss-to-text (G2T), and sign-to-gloss-to-text (S2G2T). Yin and Read (2020) converted gloss sequences extracted by CSLRs into sentences via a transformer structure. In the S2G2T model, signs are converted to glosses that are subsequently translated into text. Recently, several studies have been conducted on end-to-end translation from sign to text, bypassing the conversion to gloss. In the study by Camgoz et al. (2020), each frame extracted by the CNN served as a token for the transformer. TSPNet employs a temporal semantic pyramid using 3D-CNN to extract multiple temporal data (Li et al., 2020; Miranda et al., 2022). Zheng et al. (2020) proposed a frame stream density compression algorithm to reduce the amount of data.

In contrast, the key point-based methods extract human key points per frame using the pose estimation framework. This reduces the number of features compared to CNN-based methods and human information can be effectively incorporated. In the work of Ko et al. (2019) and Kim and Baek (2023), the key points extracted from OpenPose were normalized, and then random frame sampling was performed. These research outperformed S2T model in Camgoz et al. (2018). Kim and Kim (2023) tokenized sign language videos into action units using key points and highlighted non-manual parts such as eyes, eyebrows, and mouth. As a result, they demonstrated that a human-informed model using key points is a promising approach for characterizing sign language.

3. Methodology

Herein, we propose a token embedding framework (ACE) that efficiently represents sign language by focusing on body movements to translate sign language into sentences, as displayed in Figure 2. The proposed framework performs key point extraction, which reveals the signer's expressions. In addition, ACE tokenizes changes in posture over time within each token into frequency components.

3.1. Transforming the Frame to Angular Information

Sign language consists of both manual and non-manual signals. Manual signals indicate gloss with hand movement, while non-manual signals express grammatical and contextual features, emotions, and attitudes through facial expressions and body postures. ACE recognizes the signer's hand postures, facial expressions, and body postures from the input video frames. Angle information is then extracted from these signals to standardize a notation system for sign language, similar to phonetic symbols in spoken languages.

Key point detection in sign language. For each input video frame, ACE detects key points from a signer's four distinct body parts: the upper body, face, left hand, and right hand. More specifically, we use an open-source human pose estimation library (OpenPose) and detect 12, 70, 21, and 21 key points from each body part, respectively. Finally, we obtain a total of 124 key points, where each key point k_i is represented as its x-y coordinate in the frame (i.e., $[x_i, y_i]$). It should be noted that ACE does not extract any key points from the signer's lower body, as these have less impact on sign language translation.

Posture construction with edge-pairs. ACE represents the signer's posture as angular information between the detected key points. Let $e_{i,j}$ denote an edge vector of two key points (k_i and k_j). Given the two vectors $e_{i,j}$ and $e_{k,l}$, ACE first computes their cosine value c as

$$c = \frac{e_{i,j} \cdot e_{k,l}}{\|e_{i,j}\| \|e_{k,l}\|}, \tag{1}$$

where $\cdot$ is the inner product of the vectors and $\|e_{i,j}\|$ and $\|e_{k,l}\|$ are the magnitudes of each vector. We repeat this process for each edge pair defined in Table 1. It should be noted that the numbering of the key points in the table follows the numbering convention of OpenPose. In other words, for the i-th input video frame, we obtain a set of cosine values denoted V_i, where $V_i = \{c_1(i), c_2(i), \ldots, c_{76}(i)\}$. Finally, ACE outputs an angular information set V, which contains the cosine values obtained from each video frame (i.e., $V = \{V_1, V_2, \ldots, V_T\}$), where T is the total number of input video frames.

Notation System for Sign language Unlike spoken languages, sign language uses a spatial dimension for conveying meaning. In other words, the positions and movements of the hands, face, and body are not arbitrary. Rather, they have structured and nuanced significance. Angular information serves as a pivotal bridge between physical gestures and linguistic meaning. Moreover, it provides a systematic and quantifiable measure of spatial relations, creating a structured notation system that is similar to how phonetics serves spoken languages.

Within each frame, the cosine values of edge pairs offer a snapshot of the signer's morphology at that particular moment, encoding both manual and non-manual signals. This granular approach allows us to capture the minutiae of sign languages, including subtle shifts in hand shapes, orientations, locations, and facial expressions. When sequenced over time, this becomes a coherent narrative that is equivalent to sentences in spoken language. Our notation system omits spaces, reflecting the continuous nature of sign language gestures. By adopting this angular-based notation system, we can systematically deconstruct, analyze, and interpret the morphological patterns inherent in sign language, providing a foundation for further linguistic and computational explorations.

3.2. Token Framework in the Frequency Domain

A key concept of ACE is to generate embeddings that are effective for sign language translation, leveraging the time varying frequency characteristics of the signer's movements. To achieve this,

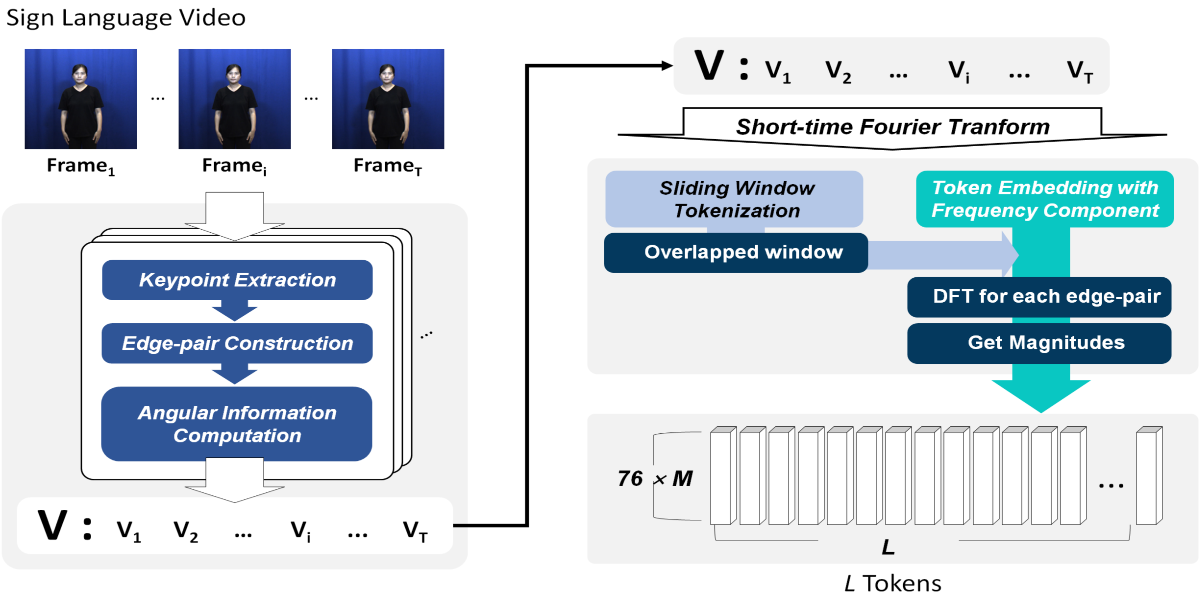

Figure 2: Overview of the ACE token embedding framework for sign language representation.

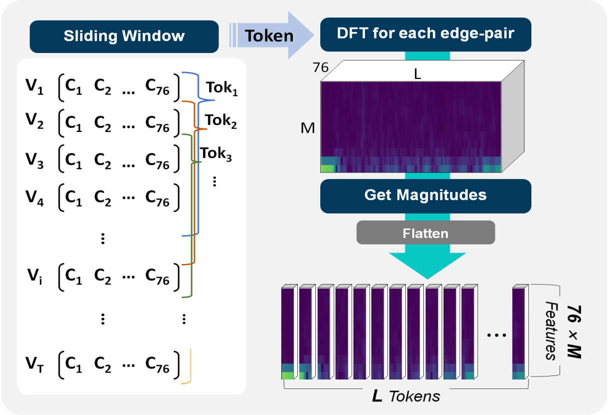

Figure 3: Illustration of tokenizing angular information using the short-time Fourier transform (STFT) method to capture time-varying frequency characteristics of signer's movements.

we apply the STFT method to the obtained angular information set V. In other words, by using STFT, we tokenize V into multiple windows and convert their frequency characteristics into token embeddings (Figure 3).

Sliding window-based tokenization. First, ACE divides the angular information set V into multiple sub-windows using a sliding window method. Each token serves as a token for sign language. Let W_i denote the i-th sub-window obtained from V, which is defined as

$$W_i = \{V_j | H\times(i-1)+1 \leq j \leq H\times(i-1)+N\}, \quad (2)$$

where N is the window size and H is the interval between two consecutive windows. H is computed as $N - O$, where O is the size of the area that overlaps with a previous window.

Token embedding with frequency components. Once tokenized, ACE extracts embeddings from each token through a frequency domain analysis. More specifically, we extract X_i, which is the one-sided fast Fourier transform (FFT) magnitudes for the i-th edge pair's angular information, as follows:

$$X_i(f) = |\sum_{k=1}^{N} c_i(k) e^{-j(2\pi fk)/N}|, \quad (3)$$

where k is the frame index in the given token. It should be noted that f ranges from 0 to $\lfloor \frac{N}{2} \rfloor - 1$ according to the Nyquist frequency theorem (Nyquist, 1928; Shannon, 1949). Subsequently, ACE produces an embedding for the token by concatenating the magnitudes of all edge pairs.

3.2.1. Insights into resolution and comprehension in sign language tokenization

It is essential to understand the properties of movement representation when analyzing sign language. Parameters N and O have a significant impact on how effectively we can capture this movement. A larger value of N can hinder the analysis of rapid movement details due to excessive frame aggregation, while a smaller value could result in insufficient context. Larger O values promote continuity, enabling analysis from multiple perspectives. To optimize sign language representation, we must investigate how N and O interact with the unique linguistic characteristics of sign language, which

Edge-pair					
No.	Edge $e_{i,j}$	Edge $e_{k,l}$	No.	Edge $e_{i,j}$	Edge $e_{k,l}$
1	BODY (0, 1)	BODY (1, 2)	2	BODY (0, 1)	BODY (1, 5)
3	BODY (2, 1)	BODY (1, 5)	4	BODY (1, 2)	BODY (2, 3)
5	BODY (2, 3)	BODY (3, 4)	6	BODY (1, 5)	BODY (5, 6)
7	BODY (5, 6)	BODY (6, 7)	8	BODY (17, 18)	BODY (0, 1)
9	BODY (6, 7)	HAND$_L$ (0, 9)	10	BODY (3, 4)	HAND$_R$ (0, 9)
11	FACE (41, 36)	FACE (36, 37)	12	FACE (40, 41)	FACE (41, 36)
13	FACE (47, 42)	FACE (42, 43)	14	FACE (46, 47)	FACE (47, 42)
15	FACE (59, 48)	FACE (48, 49)	16	FACE (58, 59)	FACE (59, 48)
17	FACE (67, 60)	FACE (60, 61)	18	FACE (66, 67)	FACE (67, 61)
19	FACE (36, 37)	FACE (37, 38)	20	FACE (37, 38)	FACE (38, 39)
21	FACE (38, 39)	FACE (39, 40)	22	FACE (39, 40)	FACE (40, 41)
23	FACE (42, 43)	FACE (43, 44)	24	FACE (43, 44)	FACE (44, 45)
25	FACE (44, 45)	FACE (45, 46)	26	FACE (45, 46)	FACE (46, 47)
27	FACE (48, 49)	FACE (49, 50)	28	FACE (49, 50)	FACE (50, 51)
29	FACE (50, 51)	FACE (51, 52)	30	FACE (51, 52)	FACE (52, 53)
31	FACE (52, 53)	FACE (53, 54)	32	FACE (53, 54)	FACE (54, 55)
33	FACE (54, 55)	FACE (55, 56)	34	FACE (55, 56)	FACE (56, 57)
35	FACE (56, 57)	FACE (57, 58)	36	FACE (57, 58)	FACE (58, 59)
37	FACE (60, 61)	FACE (61, 62)	38	FACE (61, 62)	FACE (62, 63)
39	FACE (62, 63)	FACE (63, 64)	40	FACE (63, 64)	FACE (64, 65)
41	FACE (64, 65)	FACE (65, 66)	42	FACE (65, 66)	FACE (66, 67)
43	FACE (17, 18)	FACE (18, 19)	44	FACE (18, 19)	FACE (19, 20)
45	FACE (22, 23)	FACE (23, 24)	46	FACE (23, 24)	FACE (24, 25)
47	HAND$_L$ (0, 1)	HAND$_L$ (1, 2)	48	HAND$_L$ (1, 2)	HAND$_L$ (2, 3)
49	HAND$_L$ (2, 3)	HAND$_L$ (3, 4)	50	HAND$_L$ (0, 5)	HAND$_L$ (5, 6)
51	HAND$_L$ (5, 6)	HAND$_L$ (6, 7)	52	HAND$_L$ (6, 7)	HAND$_L$ (7, 8)
53	HAND$_L$ (0, 9)	HAND$_L$ (9, 10)	54	HAND$_L$ (9, 10)	HAND$_L$ (10, 11)
55	HAND$_L$ (10, 11)	HAND$_L$ (11, 12)	56	HAND$_L$ (0, 13)	HAND$_L$ (13, 14)
57	HAND$_L$ (13, 14)	HAND$_L$ (14, 15)	58	HAND$_L$ (14, 15)	HAND$_L$ (15, 16)
59	HAND$_L$ (0, 17)	HAND$_L$ (17, 18)	60	HAND$_L$ (17, 18)	HAND$_L$ (18, 19)
61	HAND$_L$ (18, 19)	HAND$_L$ (19, 20)	62	HAND$_R$ (0, 1)	HAND$_R$ (1, 2)
63	HAND$_R$ (1, 2)	HAND$_R$ (2, 3)	64	HAND$_R$ (2, 3)	HAND$_R$ (3, 4)
65	HAND$_R$ (0, 5)	HAND$_R$ (5, 6)	66	HAND$_R$ (5, 6)	HAND$_R$ (6, 7)
67	HAND$_R$ (6, 7)	HAND$_R$ (7, 8)	68	HAND$_R$ (0, 9)	HAND$_R$ (9, 10)
69	HAND$_R$ (9, 10)	HAND$_R$ (10, 11)	70	HAND$_R$ (10, 11)	HAND$_R$ (11, 12)
71	HAND$_R$ (0, 13)	HAND$_R$ (13, 14)	72	HAND$_R$ (13, 14)	HAND$_R$ (14, 15)
73	HAND$_R$ (14, 15)	HAND$_R$ (15, 16)	74	HAND$_R$ (0, 17)	HAND$_R$ (17, 18)
75	HAND$_R$ (17, 18)	HAND$_R$ (18, 19)	76	HAND$_R$ (18, 19)	HAND$_R$ (19, 20)

Table 1: Edge pairs used for extracting the angular information of sign language. A total of 76 edge pairs were selected to encapsulate human movements.

will guide us toward answering the following fundamental questions:

- **RQ1:** To what extent does increasing overlap size O influence the ability to represent sign language from diverse analytical perspectives?
- **RQ2:** What size of window N maximizes the quality of sign language representation, balancing the need for detail and sufficient context?

In Section 4.1, we conduct experiments in different configurations, where the parameters are set to explore our two research questions. The resulting performance is then evaluated to answer the questions.

3.3. ACE-Enhanced Transformer Structure

The ACE framework is incorporated into a transformer architecture to capture the intrinsic nuances of sign languages more effectively. The robust representation of signer actions and postures feeds into the encoder layer. At the decoder layer, the model tokenizes sentences with a wordpiece-based byte-pair encoding tokenizer, which is trained on a given corpus of translation label sentences. The detailed code and the model implementation can be found in the attached GitHub[1].

4. Experiments

This section presents the experimental results that validate the performance of ACE. We explored the RQs by comparing the translation performance of the hyperparameters in ACE. Furthermore, the value of the STFT-based token embedding framework was evaluated by comparing it with previous research that used key points as token embedding.

4.1. Setup

For sign language translation, the transformer (Vaswani et al., 2017) model was built using PyTorch. The model comprises three transformer layers, each with a dimension of 256 and 4 heads. Our decoder used a vocabulary size of 22,000 and the model was trained over 50 epochs, with a dropout rate of 0.1 to avoid overfitting. The batch processing was configured to manage 32 samples with a Nvidia RTX 3090 GPU. The Adam optimizer was used with a learning rate of 0.001. For the training data, we aggregated the video data of sign language for disaster safety information from the AI Hub platform. Of the 160,677 training samples (excluding incomplete data), 95% was allocated for training and 5% for testing. The STFT was implemented using scipy (Virtanen et al., 2020) with the Hann window function, without any padding or boundary conditions.

We used the BLEU-4, ROUGE-L, and METEOR evaluation metrics to determine the translation performance. A high BLEU-4 score indicated that the machine output closely resembled that of a human, focusing on the accuracy of the text produced. In contrast, ROUGE-L emphasized the coverage or thoroughness between the translated and reference text. The METEOR metric provided a comprehensive and nuanced assessment, which included the accuracy and comprehensiveness of the translated material.

4.2. Exploring the Impact on Sign Language Translation

This section presents the experimental analytics of the RQs. We addressed the RQs by examining variations in the evaluation metrics that focused on window and overlap size. We also discuss the impact of the proposed framework on sign language translation.

- **RQ1:** To what extent does increasing overlap size O influence the ability to represent sign language from diverse analytical perspectives?

We analyzed the metrics for machine translation while progressively adjusting the overlap sizes for various window sizes. Table 2 presents the metric related to variations in overlap size O. For N values of 30, 20, and 15, the highest scores were achieved when the O values were 20, 15, and 10, respectively. Specifically, significant differences emerged between ROUGE-L and METEOR. The evaluation scores were directly proportional to the overlap area across the window sizes. Moreover, an increase in the overlap size enhanced the chances of correlating one movement to another. The ACE framework supported the model in achieving a deeper understanding of the movements, enabling a clear interpretation of the signer's utterances.

- **RQ2:** What size of window N maximizes the quality of sign language representation, balancing the need for detail and sufficient context?

Table 3 presents an analysis of the metrics as N changed while maintaining a similar overlap ratio. The BLEU-4 and ROUGE-L score appeared to be inversely proportional to increases in N, and METEOR exhibited minor deviations at values below N = 50, after which it reduced significantly.

In an additional analysis, we further explored the relationship between the window size and the translation completeness. Here, we introduced the metric of sentence information size, which was derived by multiplying the average token length with the token embedding dimension. A comparison between the sentence information size and ROUGE-L indicated that the sign language translation's ability to convey meaning was proportional to the information quantity in a sign language sentence.

The window size affected how the model represented sign language tokens. While a larger size captured more movement per token, it could oversimplify the expression of sign language sentences. The abundance of information in tokens rendered them more challenging to distinguish, resulting in reduced accuracy. Therefore, an appropriate window size should be selected based on the directionality of the translation.

The experimental results revealed significant findings regarding the translation of sign language. An intricate association between the overlap and window sizes was essential to enhance the quality and clarity of the translation. Increasing the overlap

[1] https://github.com/Splo2t/action_concentrated_embedding

N	O	Avg. Token length	BLEU-4	ROUGE-L	METEOR
30	20	44.18	36.72	40.10	41.44
30	10	22.36	32.36	32.83	38.24
30	5	17.92	29.67	30.23	36.40
30	0	15.11	26.86	26.96	33.99
20	15	89.83	38.21	40.69	42.65
20	10	45.19	36.44	39.02	41.56
20	5	30.23	33.18	34.68	38.91
20	0	22.82	30.91	31.09	37.32
15	10	90.83	38.80	41.29	42.74
15	5	45.64	36.73	39.14	41.32
15	0	30.65	32.53	34.12	38.29

Table 2: Performance metrics for various window sizes and overlap ratios.

N	Overlap Ratio	Sentence Information Size	BLEU-4	ROUGE-L	METEOR
80	75%	61,765	32.71	34.06	38.40
60	75%	65,162	35.60	36.72	40.15
50	74%	64,332	35.63	37.64	40.71
40	75%	68,924	37.00	38.90	41.60
30	76.66%	76,470	37.87	40.28	42.20
20	75%	75,090	38.21	41.00	42.79
15	73.33%	68,965	38.57	41.79	42.64

Table 3: Evaluation metrics for different window sizes while maintaining similar overlap ratios. Results are averaged over three different random seeds to account for variability.

size improved the interpretation of sign language patterns, although an equilibrium with the window size was crucial to avoid losing the meaning of the message. This research sets the foundation for further research in the field of sign language translation automation to achieve more significant improvements.

4.3. Impact of Body Part Features on ACE

In this experiment, we evaluated the influence of various body part features on the performance of ACE using a maximum sequence length of 50 and the following set of parameters: N = 40 and O = 30.

From Table 4, it is evident that when ACE employed only the "Hands" feature, its performance gap to the baseline was relatively small. Similarly, even when using both "Hands" and "Face" features, ACE's performance improvement was relatively small. However, when the experiments included the "Body" feature, ACE's performance notably surpassed the baseline. This differential in performance can be attributed to ACE's design, which optimizes angular information to capture relational dynamics between body parts. When confined to the "Hands" feature, ACE's potential diminished because "Hands" alone only offers a constrained range of angular data, overlooking intricate relationships with other body parts (such as the torso and shoulders).

In contrast, the baseline model was skillful in identifying hand-to-hand relationships that used normalized key points, rendering it less dependent on the broader body context compared to ACE. By incorporating the "Body" feature, ACE gained extensive angular relations across the body, enhancing its relational understanding. This result also highlighted the essential role of dynamic body movements in advancing sign language translation algorithms.

4.4. Comparison with Prior Work

In this research, we compared the machine translation performance of the proposed and existing models in a constrained environment. Previous research based on key points introduced random sampling on frame sequences to optimize computing resources. In this experiment, we evaluated the performance of the proposed model with various numbers of sampling tokens. Figure 4 lists the automatic translation performance metric according to various maximum token lengths. When comparing the models at a maximum of 50 tokens, the proposed model exhibited improvements of 5.43%, 4.32%, and 3.96% in the BLEU-4, ROUGE-L, and METEOR scores, respectively. When the model had only 25 max tokens, the proposed model achieved a performance improvement of over 6.42%, 5.46%, and 5.16% in the BLEU-4,

Model	Metric	Features			
		Hands	Hands, Body	Hands, Face	Hands, Body, Face
Baseline	BLEU-4	29.65	30.31	30.43	30.82
	ROUGE-L	32.87	33.08	32.9	33.58
	METEOR	36.6	37.37	37.42	37.21
ACE	BLEU-4	31.97	35.23	33.16	36.78
	ROUGE-L	36.14	38.86	36.56	38.39
	METEOR	37.54	40.44	38.65	41.51

Table 4: Influence of body part features on the performance of ACE and the baseline model (Ko et al., 2019).

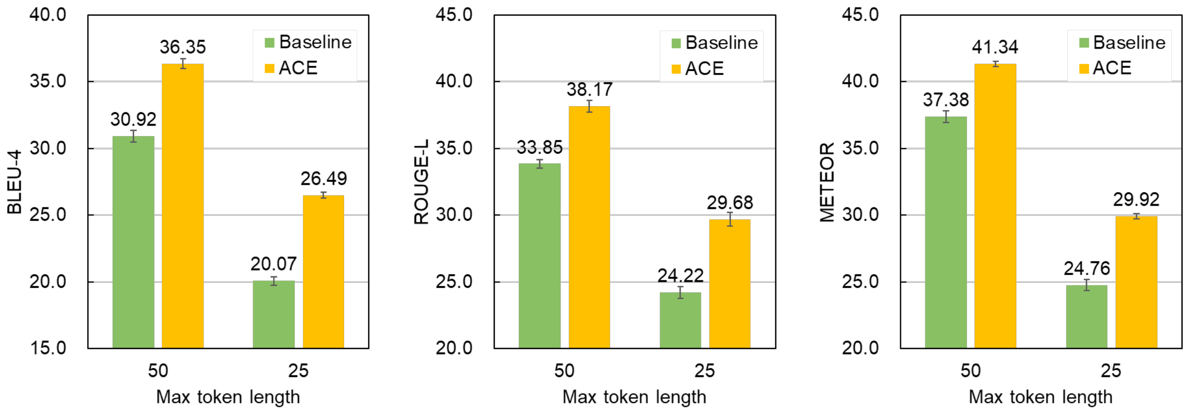

Figure 4: Comparison of automatic translation performance metrics for various maximum token lengths between the proposed ACE and the baseline by (Ko et al., 2019). Results are averaged over three different random seeds to account for variability.

ROUGE-L, and METEOR scores, respectively.

Although frame sampling only covers the instantaneous posture of the sign language, the proposed method did not have this problem because the signer's movements defined the token. Furthermore, the overlapping feature allowed the inference of lost sign language expressions when a token was sampled. Accordingly, the proposed ACE presented improved evaluation metrics for sign language performance, even with a limited token length.

5. Conclusion

In this paper, we introduced the ACE framework, which is a novel approach to sign language tokenization and understanding. By focusing on the dynamics of body movements and leveraging the STFT for tokenization, ACE displayed promising results in sign language-to-sentence translation. The experimental results confirmed ACE's adaptability and efficiency regarding the number of tokens. Especially under the restriction of a limited number of tokens, the increase in BLEU-4 and ROUGE-L metrics was remarkable. The ACE framework's innovative approach to capturing and representing sign language nuances highlighted the vast potential of bridging the communication gap between the deaf community and society. This study provided a strong and comprehensive depiction of sign language, while also establishing a solid framework for future research. Its findings bring us closer to the ultimate objective of precise and effective comprehension and translation of sign language.

6. Acknowledgement

This research was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C109550811 and NRF-2022R1C1C1012664) and in part by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0012724, The Competency Development Program for Industry Specialist).

7. Bibliographical References

Muneer Al-Hammadi, Ghulam Muhammad, Wadood Abdul, Mansour Alsulaiman, Mohamed A. Bencherif, and Mohamed Amine

Mekhtiche. 2020a. Hand gesture recognition for sign language using 3dcnn. *IEEE Access*, 8:79491–79509.

Muneer Al-Hammadi, Ghulam Muhammad, Wadood Abdul, Mansour Alsulaiman, Mohammed A. Bencherif, Tareq S. Alrayes, Hassan Mathkour, and Mohamed Amine Mekhtiche. 2020b. Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. *IEEE Access*, 8:192527–192542.

Sholikhatul Amaliya, Anik Nur Handayani, Muhammad Iqbal Akbar, Heru Wahyu Herwanto, Osamu Fukuda, and Wendy Cahya Kurniawan. 2021. Study on hand keypoint framework for sign language recognition. In *2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, pages 446–451. IEEE.

Galina Angelova, Eleftherios Avramidis, and Sebastian Möller. 2022. Using neural machine translation methods for sign language translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 273–284, Dublin, Ireland. Association for Computational Linguistics.

Ali Araabi and Christof Monz. 2020. Optimizing transformer for low-resource neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3429–3435, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Nanziba Basnin, Lutfun Nahar, and Mohammad Shahadat Hossain. 2020. An integrated cnn-lstm model for bangla lexical sign language recognition. In *Proceedings of International Conference on Trends in Computational and Cognitive Engineering: Proceedings of TCCE 2020*, pages 695–707. Springer.

Cyrille Beaudry, Renaud Péteri, and Laurent Mascarilla. 2014. Action recognition in videos using frequency analysis of critical point trajectories. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 1445–1449.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 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 (CVPR)*.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*.

Zewen Chi, Li Dong, Shuming Ma, Shaohan Huang, Saksham Singhal, Xian-Ling Mao, Heyan Huang, Xia Song, and Furu Wei. 2021. mT6: Multilingual pretrained text-to-text transformer with translation pairs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1671–1683, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yuqian Dai, Marc de Kamps, and Serge Sharoff. 2022. Bertology for machine translation: What bert knows about linguistic difficulties for translation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6674–6690.

Cleison Correia de Amorim, David Macêdo, and Cleber Zanchettin. 2019. Spatial-temporal graph convolutional networks for sign language recognition. In *International Conference on Artificial Neural Networks*, pages 646–657. Springer.

D. Griffin and Jae Lim. 1984. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243.

Jie Huang, Wengang Zhou, Houqiang Li, and Weiping Li. 2015. Sign language recognition using 3d convolutional neural networks. In *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6.

Peiqi Jiao, Yuecong Min, Yanan Li, Xiaotao Wang, Lei Lei, and Xilin Chen. 2023. Cosign: Exploring co-occurrence signals in skeleton-based continuous sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20676–20686.

Jungeun Kim and Ha Young Kim. 2023. Cslt-ak: Convolutional-embedded transformer with an action tokenizer and keypoint emphasizer for sign language translation. *Pattern Recognition Letters*, 173:115–122.

Youngmin Kim and Hyeongboo Baek. 2023. Preprocessing for keypoint-based sign language translation without glosses. *Sensors*, 23(6):3231.

Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho. 2019. Neural sign language translation based on human keypoint estimation. *Applied sciences*, 9(13):2683.

Sang-Ki Ko, Jae Gi Son, and Hyedong Jung. 2018. Sign language recognition with recurrent neural network using human keypoint detection. In *Proceedings of the 2018 conference on research in adaptive and convergent systems*, pages 326–328.

Oscar Koller, Necati Cihan Camgoz, Hermann Ney, and Richard Bowden. 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.

Oscar Koller, O Zargaran, Hermann Ney, and Richard Bowden. 2016. Deep sign: Hybrid cnn-hmm for continuous sign language recognition. In *Proceedings of the British Machine Vision Conference 2016*.

Oscar Koller, Sepehr Zargaran, and Hermann Ney. 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*, pages 4297–4305.

Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Benjamin Swift, Hanna Suominen, and Hongdong Li. 2020. Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. *Advances in Neural Information Processing Systems*, 33:12034–12045.

Diane C Lillo-Martin, Elaine Gale, and Deborah Chen Pichler. 2023. Family asl: An early start to equitable education for deaf children. *Topics in Early Childhood Special Education*, 43(2):156–166.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. Mediapipe: A framework for perceiving and processing reality. In *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019*.

Yongsen Ma, Gang Zhou, Shuangquan Wang, Hongyang Zhao, and Woosub Jung. 2018. Signfi: Sign language recognition using wifi. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(1).

Péricles BC Miranda, Vitor Casadei, Emely Silva, Jayne Silva, Manoel Alves, Marianna Severo, and João Paulo Freitas. 2022. Tspnet-hf: A hand/face tspnet method for sign language translation. In *Ibero-American Conference on Artificial Intelligence*, pages 305–316. Springer.

Rafizah Mohd Hanifa, Khalid Isa, and Shamsul Mohamad. 2021. A review on speaker recognition: Technology and challenges. *Computers Electrical Engineering*, 90:107005.

Mathias Müller, Zifan Jiang, Amit Moryossef, Annette Rios, and Sarah Ebling. 2023. Considerations for meaningful sign language machine translation based on glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 682–693, Toronto, Canada. Association for Computational Linguistics.

Harry Nyquist. 1928. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644.

M. Mahbubur Rahman, Robiulhossain Mdrafi, Ali C. Gurbuz, Evie Malaia, Chris Crawford, Darrin Griffin, and Sevgi Z. Gurbuz. 2021. Word-level sign language recognition using linguistic adaptation of 77 ghz fmcw radar data. In *2021 IEEE Radar Conference (RadarConf21)*, pages 1–6.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

C.E. Shannon. 1949. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21.

Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. 2017. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*.

Anh Tran, Jinyan Guan, Thanima Pilantanakitti, and Paul Cohen. 2014. Action recognition in the frequency domain.

Anirudh Tunga, Sai Vidyaranya Nuthalapati, and Juan Wachs. 2021. Pose-based sign language recognition using gcn and bert. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 31–40.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Lucas Nunes Vieira, Minako O'Hagan, and Carol O'Sullivan. 2021. Understanding the societal impacts of machine translation: a critical review of the literature on medical and legal use cases. *Information, Communication & Society*, 24(11):1515–1532.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. 2016. Convolutional pose machines. In *CVPR*.

Feng Wen, Zixuan Zhang, Tianyiyi He, and Chengkuo Lee. 2021. Ai enabled sign language recognition and vr space bidirectional communication using triboelectric smart glove. *Nature communications*, 12(1):5378.

Su Yang and Qing Zhu. 2017. Continuous chinese sign language recognition with cnn-lstm. In *Ninth international conference on digital image processing (ICDIP 2017)*, volume 10420, pages 83–89. SPIE.

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7347–7360, Online. Association for Computational Linguistics.

Kayo Yin and Jesse Read. 2020. Better sign language translation with stmc-transformer. *arXiv preprint arXiv:2004.00588*.

Haotian Zhang, Lukun Wang, Jiaming Pei, Feng Lyu, Minglu Li, and Chao Liu. 2023a. Rf-sign: Position-independent sign language recognition using passive rfid tags. *IEEE Internet of Things Journal*.

Hengbo Zhang, Daming Liu, and Nana Fu. 2023b. Continuous sign language recognition based on 3dcnn and blstm. In *Fifth International Conference on Computer Information Science and Artificial Intelligence (CISAI 2022)*, volume 12566, pages 505–512. SPIE.

Lei Zhang, Yixiang Zhang, and Xiaolong Zheng. 2020. Wisign: Ubiquitous american sign language recognition using commercial wi-fi devices. *ACM Trans. Intell. Syst. Technol.*, 11(3).

Jiangbin Zheng, Zheng Zhao, Min Chen, Jing Chen, Chong Wu, Yidong Chen, Xiaodong Shi, Yiqi Tong, et al. 2020. An improved sign language translation model with explainable adaptations for processing long sign sentences. *Computational Intelligence and Neuroscience*, 2020.

Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li. 2020a. Spatial-temporal multi-cue network for continuous sign language recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13009–13016.

Zhihao Zhou, Kyle Chen, Xiaoshi Li, Songlin Zhang, Yufen Wu, Yihao Zhou, Keyu Meng, Chenchen Sun, Qiang He, Wenjing Fan, et al. 2020b. Sign-to-speech translation using machine-learning-assisted stretchable sensor arrays. *Nature Electronics*, 3(9):571–578.