HamNoSys-based Motion Editing Method for Sign Language

Tsubasa Uchida, Taro Miyazaki, Hiroyuki Kaneko
NHK Science & Technology Research Laboratories
Tokyo Japan
{uchida.t-fi, miyazaki.t-jw, kaneko.h-dk}@nhk.or.jp

Abstract

We have developed a Japanese-to-Japanese Sign Language (JSL) translation system to expand sign language services for the Deaf. Although recording the motion data of isolated JSL by motion capture (MoCap) and avatar animation driven by MoCap data is effective for capturing the more natural movements of sign language, the disadvantage is that they lack the flexibility to reproduce the contextual modification of signs. We therefore propose a sign language motion data editing method based on the Hamburg Notation System for Sign Languages (HamNoSys) for use in a hybrid system that combines a MoCap data-driven technique and a phonological generation technique. The proposed method enables the editing of handshape, hand orientation, and location of the motion data based on HamNoSys components to generate contextual modifications for motion-captured citation form signs in translated gloss sequences. Experimental results demonstrate that our method achieves the flexibility to generate contextual modifications and new movements while preserving natural human-like movements without the need for additional MoCap processes.

Keywords: HamNoSys, Avatar Animation, Motion Capture, Japanese Sign Language (JSL), Classifier

1. Introduction

In order to improve accessibility for the Deaf, the number of services with sign language (SL) has been gradually increasing in recent years. When it comes to expanding SL services, translation from spoken language to SL is the most basic and important requirement for Deaf people. The best way to provide information in SL is through real-time interpretation by a human signer, and in some cases using pre-recorded video of signers. The pre-recorded video method allows for higher quality SL interpretation than real-time because it can be prepared in advance, but creating a signed video from spoken text or audio requires recording time and cost. Additionally, as human video material of interpreters signing is difficult and inflexible to reuse, signers are required to translate and record new videos each time they create new content. There is also another issue regarding the anonymization of signers in video material (Xia et al., 2022).

Therefore, several automatic translation methods from spoken language to SL have been proposed to increase the number of signed videos without human intervention. Recently, with the development of deep neural network technology, end-to-end translation methods that generate photo-realistic SL videos from input spoken language have been proposed (Saunders et al., 2020, 2022). However, even though hands and body movements are reproduced in the output video, facial expressions and mouth shape cannot be completely reproduced. At present, the mainstream output of SL translation comprises avatar animation created using computer graphics (CG) technology. There are several methods when it comes to producing an avatar animation as an output of SL translation, such as hand-crafted keyframe animation (McDonald et al., 2016), phonological-based generation (Nunnari et al., 2018; McDonald and Filhol, 2021), and data-driven methods using motion capture (MoCap) data (Gibet et al., 2011; Naert et al., 2020; Brock and Nakadai, 2018). Since the motion data-driven method can reproduce natural movements that are more realistic than other methods, we have developed a Japanese Sign Language (JSL) translation system that utilizes pre-recorded MoCap data (Miyazaki et al., 2023; Uchida et al., 2023).

SL translation systems mainly use utterance synthesis to concatenate sign words together for accurately representing sentences in spoken languages (Kim et al., 2022; Ebling, 2016; Morrissey, 2008). In utterance synthesis, it is not enough to simply concatenate an SL word in a citation form because in SL, word forms such as handshape, location, speed, and size change depending on the context. Therefore, to generate accurate SL sentences, it is essential to reproduce the modification of signs according to the context of the original spoken sentences (Naert et al., 2020). We are currently developing an editing tool to reproduce contextual modifications of JSL by modifying MoCap data captured in a citation form (Uchida et al., 2023). The term "contextual modification" is used in this paper to refer to modifications of signs based on the linguistic context in the broad sense, such as prosody, assimilation, and morphological modifications.

In this paper, we present a motion editing method

Proceedings of the 11th Workshop on the Representation and Processing of Sign Languages, pages 376–385
25 May 2024.

based on the Hamburg Notation System for Sign Languages (HamNoSys) (Hanke, 2004). Our contributions are summarized as follows:

1. We developed a motion editing method specialized for MoCap data-driven SL avatar animation.

2. Our method can edit the handshape, hand orientation, and location of pre-recorded MoCap data based on the linguistic components of HamNoSys.

3. Evaluation experiments showed that our method can output natural human-like movement and offers the flexibility to generate contextual modifications and new movement to improve the intelligibility of avatar animations without additional MoCap processes.

2. Related Work

Approaches to SL avatar animation production include hand-crafted keyframe animation, phonological-based generation, and data-driven methods. Each method has its own advantages and disadvantages, and the perfect generation method has yet to be established (Wolfe et al., 2022). All three methods mainly utilize utterance synthesis, which generates SL sentences by concatenating isolated citation form signs.

The first hand-crafted keyframe animation method generates avatar animation by interpolating the motion between the key poses of the skeleton (McDonald et al., 2016). This method has the advantage of being able to generate motions for SL sentences using a small amount of pose data, and it can flexibly respond to natural contextual modifications by creating poses with various patterns. The disadvantage of this approach is that the quality and quantity of the output animation depends on the quality and the quantity of manual labor. All poses have to be created by hand by animators in advance, requiring a large amount of work from highly skilled animators to achieve high quality and varied translations.

The second phonological-based generation method can generate motions that reproduce contextual modifications in SL utterances by taking linguistic information into account. Some researchers are developing a system that parametrically generates avatar motion by combining phonological components based on SL description methods proposed through SL linguistic research, such as Stokoe et al. (1965); Hanke (2004).

This approach allows the system to freely generate motions by selecting and combining multiple phonological parameters, so there is no need to prepare motion data in advance. It is also extremely flexible for reproducing all contextual modifications in an utterance (Elliott et al., 2007; Fotinea et al., 2007; McDonald and Filhol, 2021). However, there is a problem in that the generated motions are typically robotic and unnatural. Furthermore, in order to use this animation generation approach in an SL translation system, there is another problem in that very detailed annotations by linguistic experts who are familiar with the structure of each description method are required when preparing training data in advance.

The final method utilizes MoCap data to manipulate avatars for generating realistic human movements. Since this method captures human movements as motion data, it can reproduce the most natural human-like SL movements among the three methods discussed here. However, MoCap recording requires large-scale studio equipment and post-processing of recorded data by experts, making this approach expensive and time-consuming. Additionally, it is difficult to reproduce contextual modifications using only motion data recorded in citation form in utterance synthesis, and it is also impossible to record the almost infinite number of SL modifications in all the possible contexts as motion data.

Therefore, some researchers have proposed hybrid methods that combine the advantages of these methods.

Huenerfauth et al. (2015) proposed modeling methods for the construction of a parameterized lexicon of ASL verb signs. They modeled the signer's hand locations and orientations during each ASL indicating verb, dependent upon the location in the signing space where the subject and object were positioned, and utilized SL MoCap data from native signers as a training data set for learning the models.

Filhol and McDonald (2018) proposed a hybrid system of the hand-crafted keyframe animation system Paula and SL description model AZee. The system can produce avatar animations that can be modified procedurally thanks to Inverse Kinematics (IK) solvers in order to synthesize proforms or spatial referencing mechanisms for utterance generation. Furthermore, Nunnari et al. (2018) proposed a bottom-up procedural computation method for the AZee animation system, allowing for the animation of different communicative channels that are interleaved on the timeline.

Gibet et al. (2011) and Naert et al. (2021) proposed a hybrid method that combines the naturalness of MoCap data-driven methods with the flexibility of the phonological-based generation method. In this method, MoCap data divided into several element channels, (e.g., handshape, position, movement, posture, and facial expression) is registered as a MoCap data corpus in advance, and the data is combined to create new SL utterances. However, this approach requires building a semantic database that serves as a mapping between the SL gloss level annotations and the movements in the MoCap data corpus, and it requires a very extensive pre-process (such as detailed segmentation) on the recorded MoCap data.

Although these methods have potential as elemental technologies for SL avatar animation production, few efforts have been made to utilize them as part of a translation system. Therefore, we propose a system that combines a MoCap data-driven method and a flexible phonological-based generation method as a motion editing tool for translation systems. The proposed system reproduces natural contextual modifications for the avatar animation that is the output of the translation system by editing the SL movement of MoCap data using HamNoSys-based phonetic units. Furthermore, it is also possible to generate new SL motions from existing MoCap data without any high-cost MoCap processes.

3. Data Preparation

3.1. Motion capture

We captured over 8,000 isolated JSL motion data by MoCap. Among the various motion capture methods that have been proposed, we opted to use an optical motion capture system, which is expected to have high accuracy. In this system, multiple markers coated with retroreflective material are attached to the signer, and reflected light is recorded utilizing multiple infrared cameras equipped with infrared LEDs around the lenses. The motion of the signer is calculated by measuring the three-dimensional position of each marker based on the recorded video. We used 42 Vicon T160 and V16 cameras and 112 retro-reflective markers on the signer's body (Table 1), which have less physical restraint than sensor gloves (Figure 1). We used significantly more cameras and markers than in general movie production shooting, because in SL, details such as whether the ring formed by the fingers is closed or not, and whether adjacent fingers are touching each other are important for understanding. Of course, in order to record non-manual markers (NMMs) such as facial expressions, which are important in JSL as well, we also recorded motion data by attaching markers to characteristic parts of the signer's face. For the handshape and facial expression, we used markers sized 3 mm in diameter because they need to be measured with high precision. Most of this data was captured as citation forms listed in the JSL dictionary, and any data loss due to marker occlusion, which is a weak point of the optical motion capture system, was carefully corrected during post-processing. The MoCap data was recorded in Filmbox (FBX) format at 120fps and post-processed into Biovision Hierarchy (BVH) format files at 60fps. BVH is a MoCap file format developed by Biovision, and is written in ASCII format, making it easy to manipulate the file contents on a computer.

In addition, we constructed a JSL motion database that can be utilized as an independent API. The database stores motion data and JSL gloss pair information. The translation system reads each linked motion data from the motion database based on the JSL gloss string that is the translation result. The motion database also provides start and end frame information for each motion data, so unnecessary frames can be cut when using utterance synthesis.

Figure 1: Optical motion capture system and markers for the JSL recording.

	Retro-reflective markers	
Body region	Diameter [mm]	Number
Face	3	33
Hands	3	24 × 2
Body	10	31
Total number		112

Table 1: Our motion capture marker set.

3.2. Bone structure of avatar

We created a CG avatar with a skeletal structure that can appropriately reproduce the obtained MoCap data (Kaneko et al., 2010). Our avatar has a total of 162 joints (one root joint, 111 body and facial joints, and 50 end-effectors). In defining the skeletal structure, we made it possible for the fingers to make fine movements at the base of the thumb and palm in addition to the joints of each finger, and for the body, we added joints to the collarbone to enable movement around the shoulders (Figure 2).

One technique for expressing facial expressions using CG is to define the facial muscles under the skin of a CG avatar's face and the facial skin surface that are linked to these as a physical model, but creating a facial muscle model is extremely difficult and complex. As an alternative, we implemented a method that controls the skin shape of a CG avatar using data that digitally records the movement of the facial skin shape using optical motion capture data. Even though it has more controllable points (joints) than the facial muscle model, it does not require simulation using a physical model, it reduces the computational load during CG animation generation, and it is easy to create a CG avatar's face.

Figure 2: Our avatar representation driven by MoCap data.

4. Proposed Method

4.1. Overview

Figure 3 shows an overview of the avatar-based Japanese-to-JSL translation system we developed. Our system translates Japanese text into a JSL gloss sequence by means of a transformer model (Miyazaki et al., 2023), then performs utterance synthesis by concatenating JSL motion data, and finally displays it as a JSL avatar animation video on a player developed on the Unity game engine platform (Unity Technologies, Accessed: 2024-02-29). We also developed a motion editing tool on the Unity platform as a function linked to the motion blending process shown in Figure 3 (Uchida et al., 2023). This tool can fix translation errors by manually replacing motion data, adjusting the motion speed and connection interval, etc. before performing utterance synthesis. We added HamNoSys-based editing, which is the method proposed in this paper, to reproduce contextual modifications from citation form MoCap data as one of the new functions of this motion editing tool.

HamNoSys is the notation system developed by the University of Hamburg as a means of transcribing signs on a phonetic level. It transcribes manual postures and movements in signs based on four components (handshape, hand orientation, location, and action), and the latest version, HamNoSys 4.0, also takes into account NMMs such as eye gaze, facial expressions, and mouth gestures (Hanke, 2004). It is a sufficiently general model of sign language phonetics that all sign languages can be transcribed. Therefore, it is also possible to create motions for other sign languages by using our proposed method and JSL motion data.

As a first stage of implementation, we have adopted three components of HamNoSys, namely, handshape, hand orientation, and location, for the motion editing function. Since there are many variations of SL actions, we implemented our proposed method by specifying the motion data that is the source of editing, using that movement as an action, and applying the other three components. The proposed method can be used not only for contextual modification editing in a JSL translation system but also as an individual tool for generating new motion data without any MoCap process. We also developed a user-friendly GUI for non-experts of HamNoSys transcription rules, where operators can intuitively select the handshape, hand orientation, and location by referring to illustrations.

Each of the three functions (handshape, hand orientation, and location) are explained below along with editing examples.

4.2. Handshape

The handshape function replaces the avatar's handshape in the motion being edited with handshape motion data prepared in advance. All rotation information of the joints from the wrist of the avatar onwards in the motion data to be edited is replaced with the rotation information of the corresponding joints of the prepared handshape motion. We prepared several types of handshape motions based on HamNoSys's handshape chart. The basis posture data for the handshape was created by selecting a motion that included the relevant handshape from the existing JSL motion database, and cutting out the keyframes. Figure 4 shows the handshape list GUI and an example of generating a JSL motion [TOKYO TOWER] by replacing only the handshape from a JSL motion [SKYTREE] using the proposed method. The operator can select which handshape to replace from the handshape illustration list.

4.3. Hand orientation

The hand orientation function rotates the avatar's wrist to change the orientation of the palm. This function rotates the palm by changing the value of the avatar's wrist joint rotation data according to the rotation angle in eight patterns defined by HamNoSys. Also, the function can change which direction to use as the rotation axis from the 18 defined directions when rotating the palm. Figure 5 shows the hand orientation GUI and an example of generating a JSL motion [EMAIL] by changing only hand orientation from a JSL motion [PAY] using the

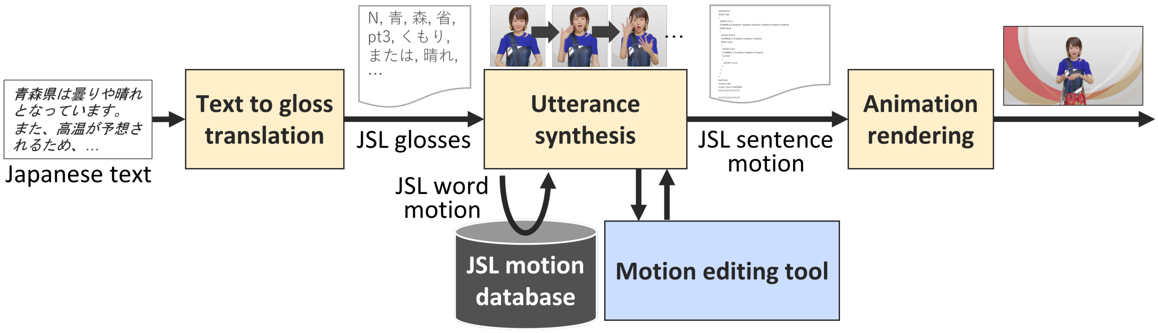

Figure 3: Japanese-to-JSL translation system.

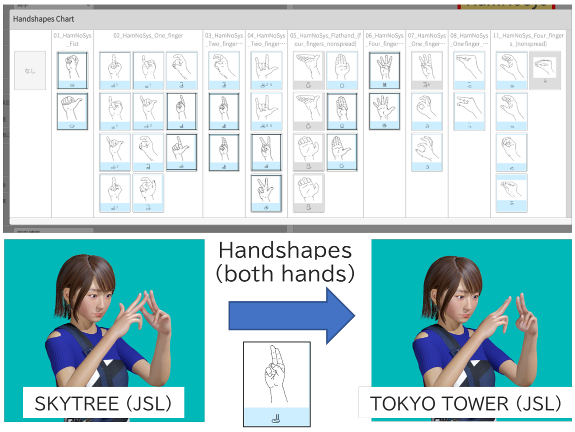

Figure 4: Handshape replace function.

proposed method. The operator can select which combination of direction and orientation to change from the hand direction and orientation illustration.

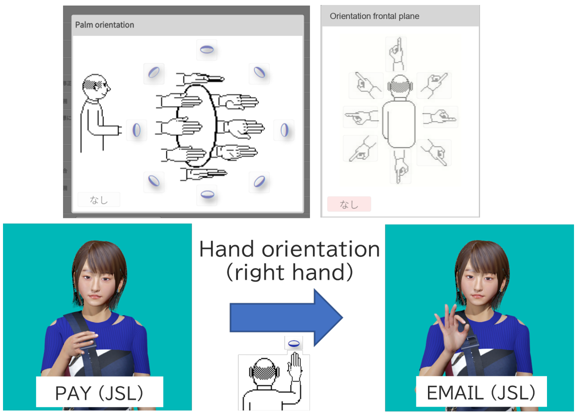

Figure 5: Hand orientation control function.

4.4. Location

The location function changes the location of the expression of signs. This function moves the location of signs to a specific location defined by HamNoSys by changing the rotation information of the avatar's arms using IK.

We used an IK articulated chain for the sign's location change. The chain consists of four joints (shoulder, elbow, forearm, and wrist) for each arm. We initially used Cyclic Coordinate Descent Inverse Kinematics (CCDIK) as an IK solver, but the hands could not reach the target position after the change process. Naert et al. (2021) proposed using Forward And Backward Reaching Inverse Kinematics (FABRIK) (Aristidou and Lasenby, 2011) for modification of hand placement. In contrast to CCDIK, which is a method that fixes the root joint position and iteratively updates the joint rotation, FABRIK is a heuristic method that obtains an IK solution by repeatedly adjusting joint angles while alternately using the root and end-effector as reference points. Figure 6 shows the difference in location change between the two types of IK solvers. By replacing CCDIK with FABRIK, the avatar's arms could be extended and reach closer to the target after the location change. Therefore, we adopted FABRIK as the new IK solver. Of course, each IK algorithm has its advantages and disadvantages, so we believe that continued consideration is necessary.

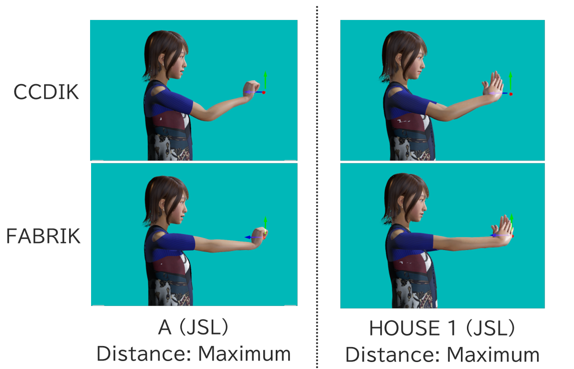

Figure 6: Comparison of IK solvers: CCDIK (upper) and FABRIK (lower).

Furthermore, depending on the original SL motion to be edited, a problem arises in that the linguistically meaningful hand configuration before editing collapses after the location is changed. This is because the rotation of the wrist is linked to changes in the posture of the arms.

Therefore, we developed a new method to reproduce the original hand configuration represented in citation form after the location change. This method offsets the rotation value of the wrists by rotating the wrists in the opposite direction after movement in accordance with the amount of rotation of the joints of the arms, thereby reproducing the configuration of the signs before editing. Figure 7 shows before and after images of applying the wrist offset method.

Figure 7: Before and after applying wrist offset method. Left: Citation form. Middle: No offset. Right: Offset.

By incorporating FABRIK and the wrist offset method, the range of contextual modifications of signs that can be reproduced by the location change function has been expanded. Figure 8 shows the location GUI and an example of contextual modification of a JSL motion [HOUSE 1] by changing its location using the proposed method. The operator can change the location of signs by choosing the illustration of the location relative to the avatar's body and face.

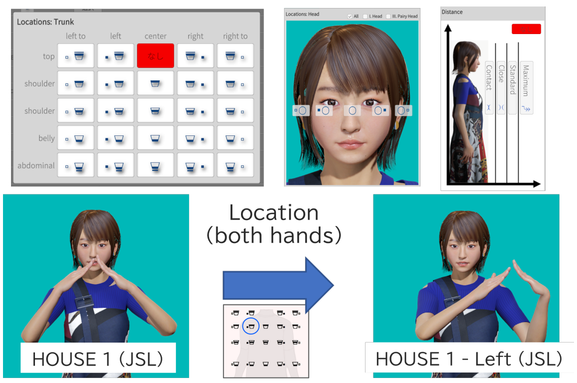

Figure 8: Location change function.

4.5. Example of motion editing by proposed system

An example of contextual motion modifications is shown in Figure 9. The GUI of the motion editing tool has a function to visually connect isolated motion data as nodes and edit the parameters of each motion data, and we also added a node dedicated to HamNoSys editing. By connecting the HamNoSys editing node to the motion data to be edited, each of the three components—handshape, hand orientation, and location—can be edited independently.

Using the proposed method, we replaced the right handshape of JSL [GO 4] and rotated the wrist in the translated JSL gloss sequence to produce a more natural JSL animation that clearly expresses the means of going and the number of people.

This usage is linguistically called a classifier (CL) predicate, and is one of the contextual modifications that can be reproduced using the proposed method. Some researchers have also worked on reproducing this CL predicate in avatar animation (Huenerfauth, 2006; Filhol and McDonald, 2020; Naert et al., 2021). Since our method is based on MoCap data, it is possible to reproduce more realistic CL predicate motions in JSL translation results than a method that uses only a phonological-based generation technique.

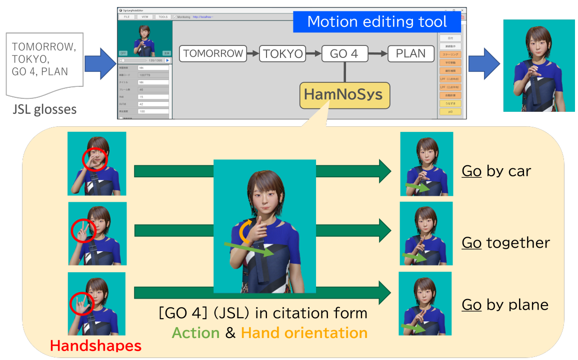

Figure 9: Example of contextual motion modifications for CL predicate.

An example of generating a German Sign Language (DGS) motion from JSL motion using our proposed method is shown in Figure 10. The upper part of the figure is an example of generating DGS motion [WICHTIG 1] by replacing only handshape from a JSL motion [STUDY 2], and the lower part is an example of generating DGS motion [SAGEN 1] by changing only hand orientation from a JSL motion [SAY 1]. As demonstrated in this example, it is also possible to generate new motions in other sign languages from JSL motions.

5. Evaluation

5.1. Design of the evaluation

We conducted an evaluation experiment on JSL avatar animations generated by our JSL translation

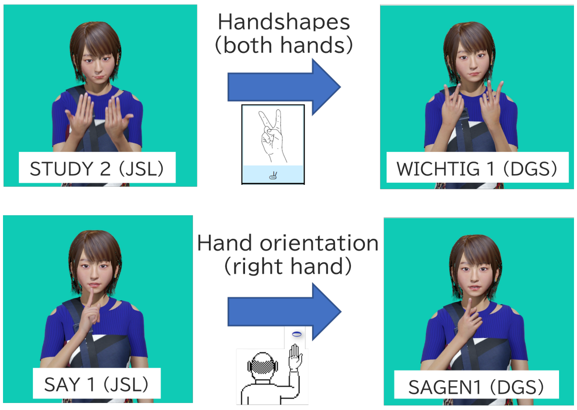

Figure 10: Example of DGS motion generation from JSL motion.

system implemented using the proposed editing method based on HamNoSys. We recruited four participants, three men and one woman, for this experiment. Two of the participants were born Deaf, one was hard of hearing, and one was a child of Deaf adults (CODA).

To investigate the effect of modifying that combines two functions, namely, handshape and hand orientation, we prepared and compared JSL avatar animations with and without modifying for JSL sentences containing CL predicates. To generate the videos used for evaluation, we selected ten Japanese news sentences that include CL predicate expressions from our Japanese-JSL news corpus. We prepared a total of 20 avatar animation videos: ten that were automatically generated by inputting the ten selected Japanese news sentences into our translation system, and ten that were manually modified using our proposed method after being automatically generated. The modification was carried out by replacing only the handshape and changing the hand orientation for the citation form motion of the word corresponding to the CL predicate. An example of modification is to change the handshape and hand orientation for the motion data of CL predicates such as [GO 4] (as shown in Figure 9), [MEET 7], [HELPED 1], and [PROTECT 1]. By modifying the JSL expressions to match the means and number of people in the original Japanese context, we aimed to clarify the subject in the JSL sentences and improve understanding of the content.

All participants evaluated all 20 avatar animation videos in the experiment. The number of video views was unlimited. The videos were presented in random order, regardless of whether they were modified by the proposed method.

Participants answered three questions after watching each video: a question testing the intelligibility of the JSL sentence, a question on the accuracy of the JSL expression, and a question about the realism of the utterance synthesis produced. All questions and answers were conducted through JSL by a JSL interpreter for Deaf and hard of hearing persons, and directly in spoken Japanese for the CODA person.

First, to check the intelligibility of the JSL sentences, we asked the subjects: “Please tell us what you were able to know by watching the animation.” This was done to determine whether they understood the JSL expressions related to CL predicates correctly in context. The second and third questions were based on questions used in the evaluation of previous studies (Naert, 2020). The second question concerned the accuracy of the JSL: “Do you think that the sign was done correctly?”, and the third question evaluated the naturalness of the movement: “Do you think that the sentence in JSL is natural/realistic/spontaneous (does it seem like the movement of a real person)?”. Both questions were answered on a 5-point Likert scale ranging from 1 (most negative) to 5 (most positive), as in previous studies.

5.2. Results

We defined the recognition rate as the percentage of people who correctly understood the meaning of the CL predicate part in each JSL sentence according to the context, and is the average value of the four participants. Figure 11 shows the recognition rate of the CL predicate part for each JSL sentence. Out of a total of ten sentences, the recognition rate of four sentences was improved after modifying by the proposed method (SL3, SL4, SL6, SL9), 3 sentences remained unchanged (SL1, SL5, SL10), and the remaining three sentences could be recognized correctly with or without modifying (SL2, SL7, SL8).

For example, in SL3, the handshape with the thumb up in the JSL for [GO 4] automatically generated by the translation system was manually replaced with a handshape with two fingers raised, making it more clear that two people are going. Similarly, in SL4, the handshape with the index finger raised in the JSL for [GO 2] was replaced with a handshape with three fingers raised, making it more clear that three people are going. Also, in SL9, the handshape representing the CL of the airplane in the JSL for [LANDING 1] was replaced with the handshape of the CL representing the train, and the participants could understand that the person arrived by train instead of by plane.

Table 2 lists the average accuracy and realism scores for each JSL sentence. Note that, the scores in Table 2 are not limited to the CL predicate part of the JSL sentence shown in Figure 11, but are scores for the entire JSL sentence. Regarding both accuracy and realism, there was no significant difference between whether or not the sentence had

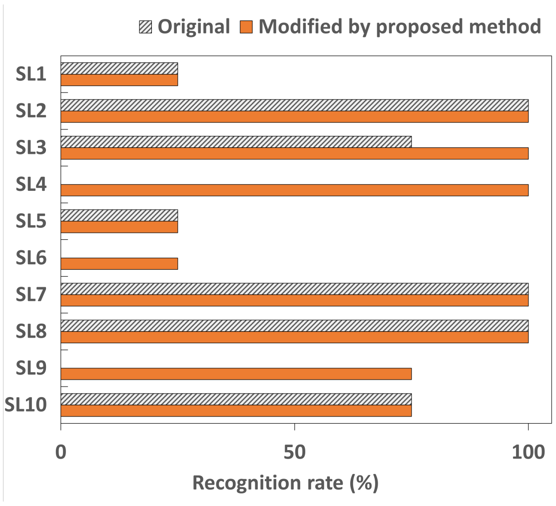

Figure 11: Recognition rate of the CL predicate part for each JSL sentence.

been modified by our proposed method. In other words, it was shown that by editing the MoCap data modified by the proposed method, contextual modifications can be reproduced without adversely affecting the quality of the data. Regarding SL10, there was no difference in recognition rate between original and modified sentence, but both accuracy and realism were improved. This is presumably due to the effect of modifying the handshape for JSL, which was expressed in the translation result as the handshape representing the CL of the car, to a handshape representing the CL of the bicycle to match the context. Also, interviews with participants revealed that three out of four were able to understand from the context that the car's CL was incorrect in the original video during the experiment.

SL	Accuracy		Realism	
	Original	Modified	Original	Modified
SL1	2.50	2.75	3.25	3.00
SL2	3.50	3.50	3.75	3.50
SL3	3.0	2.75	3.00	2.75
SL4	3.75	3.00	3.75	3.50
SL5	3.50	3.00	3.50	3.50
SL6	3.50	3.00	3.00	3.00
SL7	3.50	3.50	3.00	3.50
SL8	3.00	3.50	3.25	3.50
SL9	3.50	3.00	2.75	3.25
SL10	2.75	3.25	2.75	3.25
Mean	3.25	3.13	3.20	3.28

Table 2: Average accuracy and realism scores for each JSL sentence.

These evaluation results demonstrate that our method achieves the flexibility to generate contextual modifications and new movements while preserving the quality of natural human-like movements without the need for additional MoCap processes. In our experiment, there were no significant differences in the evaluation results between the Deaf, hard of hearing, and CODA persons, but as a future challenge, we need to confirm the reproducibility of the proposed method's effectiveness by increasing the number of participants.

6. Conclusion

In this paper, we presented our HamNoSys-based sign language motion data editing method. This method is a hybrid that combines two utterance synthesis methods: a MoCap data-driven method and a phonological-based generation method. We implemented this method in the motion editing tool of our JSL translation system and confirmed that it is possible to edit the citation form of signs included in the JSL gloss string of the translation results as CL predicates. Our evaluation experiment revealed that by applying motion modification to the translation results using the proposed method, the intelligibility of the JSL avatar animation was improved. The proposed method achieved both natural human-like movements and the flexibility to generate contextual modifications and new movements without any additional MoCap processes. Additionally, since HamNoSys supports the transcription of all sign languages, it is also possible to create motions for other sign languages by using our JSL motion data.

In future work, we plan to investigate ways of supporting other contextual modifications such as directional verbs by considering the action component of HamNoSys. We will also explore supporting NMMs such as facial expressions and mouth gestures, which are semantically important components of SL.

7. Acknowledgements

We wish to thank Thomas Hanke, Maria Kopf and the researchers at the Institute of German Sign Language and Communication of the Deaf at the University of Hamburg for their valuable inputs and assistance with the explanation of HamNoSys provided in our work. We would also like to thank the members of the Intelligent Automatic Sign Language Translation (EASIER) project for their valuable suggestions and sharing information about their latest research.

8. Bibliographical References

Andreas Aristidou and Joan Lasenby. 2011. Fabrik: A fast, iterative solver for the inverse kinematics problem. *Graphical Models*, 73(5):243–260.

Heike Brock and Kazuhiro Nakadai. 2018. Deep JSLC: A multimodal corpus collection for data-driven generation of Japanese Sign Language expressions. In *11th International Conference on Language Resources and Evaluation (LREC 2018)*, pages 4247–4252, Miyazaki, Japan. European Language Resources Association (ELRA).

Sarah Ebling. 2016. Automatic translation from german to synthesized swiss german sign language. *PhD Thesis (University of Zurich)*.

Ralph Elliott, John R. W. Glauert, Richard Kennaway, Ian Marshall, and Éva Sáfár. 2007. Linguistic modelling and language-processing technologies for avatar-based sign language presentation. *Universal Access in the Information Society*, 6(4):375–391.

Michael Filhol and John C. McDonald. 2018. Extending the AZee-Paula shortcuts to enable natural proform synthesis. In *Proceedings of the LREC2018 8th Workshop on the Representation and Processing of Sign Languages: Involving the Language Community*, pages 45–52, Miyazaki, Japan. European Language Resources Association (ELRA).

Michael Filhol and John C. McDonald. 2020. The synthesis of complex shape deployments in sign language. In *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*, pages 61–68, Marseille, France. European Language Resources Association (ELRA).

Stavroula-Evita Fotinea, Eleni Efthimiou, George Caridakis, and Kostas Karpouzis. 2007. A knowledge-based sign synthesis architecture. *Universal Access in the Information Society*, 6(4):405–418.

Sylvie Gibet, Nicolas Courty, Kyle Duarte, and Thibaut Le Naour. 2011. The signcom system for data-driven animation of interactive virtual signers: methodology and evaluation. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 1(1):1–23.

Thomas Hanke. 2004. HamNoSys – representing sign language data in language resources and language processing contexts. In *Proceedings of the LREC2004 Workshop on the Representation and Processing of Sign Languages: From SignWriting to Image Processing. Information techniques and their implications for teaching, documentation and communication*, pages 1–6, Lisbon, Portugal. European Language Resources Association (ELRA).

Matt Huenerfauth. 2006. Generating american sign language classifier predicates for english-to-asl machine translation. *PhD thesis (University of Pennsylvania)*.

Matt Huenerfauth, Pengfei Lu, and Hernisa Kacorri. 2015. Synthesizing and evaluating animations of american sign language verbs modeled from motion-capture data. *Proceedings of the 6th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT 2015)*, pages 22–28.

Hiroyuki Kaneko, Narichika Hamaguchi, Mamoru Doke, and Seiki Inoue. 2010. Sign language animation using tvml. *Proceedings of the 9th ACM SIGGRAPH Conference on Virtual-Reality Continuum and its Applications in Industry (VRCAI '10)*, pages 289–292.

Jung-Ho Kim, Eui Jun Hwang, Sukmin Cho, Du Hui Lee, and Jong Park. 2022. Sign language production with avatar layering: A critical use case over rare words. In *13th International Conference on Language Resources and Evaluation (LREC 2022)*, pages 1519–1528, Marseille, France. European Language Resources Association (ELRA).

John C. McDonald and Michael Filhol. 2021. Natural synthesis of productive forms from structured descriptions of sign language. *Machine Translation*, 35(4):1–24.

John C. McDonald, Rosalee Wolfe, Jerry Schnepp, Julie Hochgesang, Diana Gorman Jamrozik, Marie Stumbo, Larwan Berke, Melissa Bialek, and Farah Thomas. 2016. An automated technique for real-time production of lifelike animations of american sign language. *Universal Access in the Information Society*, 15(4):551–566.

Taro Miyazaki, Tsubasa Uchida, Naoki Nakatani, Hiroyuki Kaneko, and Masanori Sano. 2023. Machine translation to sign language using post-translation replacement without placeholders. *2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*.

Sara Morrissey. 2008. Data-driven machine translation for sign languages. *PhD Thesis (Dublin City University)*.

Lucie Naert. 2020. Capture, annotation and synthesis of motions for the data-driven animation of sign language avatars. *PhD Thesis (Southern Brittany University)*.

Lucie Naert, Caroline Larboulette, and Sylvie Gibet. 2020. LSF-ANIMAL: A motion capture corpus in French Sign Language designed for the animation of signing avatars. In *12th International Conference on Language Resources and Evaluation (LREC 2020)*, pages 6008–6017, Marseille, France. European Language Resources Association (ELRA).

Lucie Naert, Caroline Larboulette, and Sylvie Gibet. 2021. Motion synthesis and editing for the generation of new sign language content: Building new signs with phonological recombination. *Machine Translation*, 35(3):405–430.

Fabrizio Nunnari, Michael Filhol, and Alexis Heloir. 2018. Animating AZee descriptions using off-the-shelf IK solvers. In *Proceedings of the LREC2018 8th Workshop on the Representation and Processing of Sign Languages: Involving the Language Community*, pages 155–162, Miyazaki, Japan. European Language Resources Association (ELRA).

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. 2020. Progressive transformers for end-to-end sign language production. *European Conference on Computer Vision (ECCV)*, pages 687–705.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. 2022. Signing at scale: Learning to co-articulate signs for large-scale photo-realistic sign language production. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5141–5151.

William C. Stokoe, Dorothy C. Casterline, and Carl G. Croneberg. 1965. *A dictionary of American sign language on linguistic principles*. Gallaudet College Press.

Tsubasa Uchida, Naoki Nakatani, Taro Miyazaki, Hiroyuki Kaneko, and Masanori Sano. 2023. Motion editing tool for reproducing grammatical elements of japanese sign language avatar animation. *2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*.

Unity Technologies. Accessed: 2024-02-29. *https://unity.com/*.

Rosalee Wolfe, John C. McDonald, Thomas Hanke, Sarah Ebling, Davy Van Landuyt, Frankie Picron, Verena Krausneker, Eleni Efthimiou, Evita Fotinea, and Annelies Braffort. 2022. Sign language avatars: A question of representation. *Information*, 13(4):206.

Zhaoyang Xia, Yuxiao Chen, Qilong Zhangli, Matt Huenerfauth, Carol Neidle, and Dimitris Metaxas. 2022. Sign language video anonymization. In *Proceedings of the LREC2022 10th Workshop on the Representation and Processing of Sign Languages: Multilingual Sign Language Resources*, pages 202–211, Marseille, France. European Language Resources Association (ELRA).