Journal of Information Systems and Informatics
Vol. 7, No. 1, March 2025 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v7i1.992 Published By DRPM-UBD

Real-Time Sign Language Recognition and Translation in Humanoid Robots Using Transformer-Based Model with a Knowledge Graph

Erick Busuulwa[1], Li-Hong Juang[2,*]

[1,2]School of Artificial Intelligence, Nanjing University of Information Science and Technology, Nanjing, 210044, Jiangsu, China.
Email: [1]busuulwaerick@gmail.com, [2,*]lipuu@qq.com

Abstract

For millions of deaf-mute individuals, sign language is the only means of communication; this creates barriers in daily interactions with non-signers, leading to the exclusion of these individuals in many areas of daily life. To address this, we propose a real-time sign language translation system using a Transformer model enhanced with a knowledge graph, designed for Human-Robot Interaction (HRI) with NAO robots. Our system bridges the communication gap by translating gestures into natural language (text). We used the RWTH-PHOENIX-Weather 2014T dataset for initial training, achieving a BLEU score of 29.1 and a Word Error Rate (WER) of 18.2% surpassing the baseline model. Due to the domain shift between human gestures and NAO robot gestures, we created a NAO-specific dataset and fine-tuned the model using transfer learning to accommodate an adapted environment and kinematic constraints that do not match the environment in which the robot was deployed. This reduced the WER to 17.6% and increased the BLEU score to 29.9. We tested our model's capability with dynamic and practical HRI scenarios through comparative experiments in Webots. Integrating a knowledge graph into our model improved contextual disambiguation, significantly enhancing translation accuracy for gestures that weren't clear. Through effectively translating gestures into natural language, our system demonstrates strong potential for practical robotic applications that promote social accessibility.

Keywords: Sign Language Translation, Human-Robot Interaction, NAO Robot, Transformer Model, Gesture Recognition, Knowledge Graph

1. INTRODUCTION

Research shows that over 420 million people in the world have hearing loss problems, 34 million of these individuals are children, and most live in low- and middle-income countries [1]. Therefore, bridging the communication gap between non-signers and deaf or hard-of-hearing people remains one of the most important social challenges. For many members of the Deaf community, sign language is their primary means of communication however, its limited general population adoption creates barriers to accessibility and social integration [2]. Developments in sign

language translation (SLT) and recognition (SLR) systems have become increasingly popular in computer vision and robotics to fill this gap.

From early sensor-based techniques that used wearable technology to record hand orientations and movements [3], [4], to vision-based techniques that use raw RGB data for recognition, Sign Language Recognition (SLR) has undergone significant development. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs), two contemporary deep learning techniques, have shown impressive performance gains [5], [6] since the advent of large-scale datasets like RWTH-PHOENIX-Weather-2014T [7]. However, in real-world applications, these approaches frequently struggle to capture temporal dependencies that are essential for dynamic gesture recognition [8], [9].

Natural language processing (NLP) and computer vision tasks have been transformed by recent developments in Transformer-based architectures, with applications extending to Sign Language Recognition (SLR) and Sign Language Translation (SLT). Transformers, like the Video Vision Transformer (ViViT), are well suited for tasks involving dynamic gestures because they have demonstrated the capacity to model long-range dependencies in video data [10], [11]. Notwithstanding these developments, kinematic limitations and domain shifts between human and robotic gestures present new difficulties when integrating Sign Language Recognition (SLR) systems into robots, like NAO robot [12].

The evolution of sign language recognition started from sensor-based approaches that used gloves and Leap Motion controllers to record precise hand movements [13], [14], [15], [16], [17], to vision-based systems that only use camera feeds [3], [8], [18], [19], sign language recognition systems have advanced. Vision-based SLR methods can be divided into two categories: dynamic gesture recognition, which uses methods like Hidden Markov Models (HMMs) and dynamic time wrapping [20], [21], and static gesture recognition, which uses algorithms like K-Nearest Neighbors [22], deep learning's introduction has greatly improved SLR. In order to capture temporal and spatial features in continuous sign language videos, CNNs and RNNs have become widely used. For better dynamic gesture recognition, for instance, [5] suggested a hybrid CNN-HMM model, and [23] , [24]combined CNNs with bi-directional LSTMs. Even with these developments, cross-modal alignment between textual and visual modalities is still a common problem for traditional architectures [6].

The ability of transformer-based architectures [11] to accurately model temporal and spatial dependencies has made them a breakthrough in Sign Language Recognition and Sign Language Translation. ViViT, a Transformer designed specifically for videos, demonstrated notable advancements in gesture recognition through the use of multi-head attention mechanisms [10]. Similarly, by combining

attention mechanisms and sequence generation, the Progressive Transformer showed success in converting text into sign language gestures and poses [25]. For tasks requiring cross-modal alignment, these techniques perform better than conventional CNNs and RNNs.

The possibilities for HRI applications have increased with the integration of social robots and SLR systems. Real-time gesture recognition and translation, for instance, has been made possible by NAO and Pepper robots, allowing for organic human-robot interaction [12], [26]. Kinematic limitations and domain shifts between human and robotic gestures are two difficulties that arise when models are adapted to robotic platforms [27]. Robots can now accurately perform sign language gestures thanks to techniques like motion retargeting and domain adaptation [4].

Even though earlier research in SLR produced impressive results, there are still many unanswered questions. First of all, most current systems ignore the complexity of dynamic sign language sequences in favor of static or isolated gestures [6], [10]. Secondly, there hasn't been enough research done on real-time performance in real-world situations, especially in HRI scenarios using Transformers. Thirdly, the use of knowledge graphs for contextual understanding mechanisms is still relatively new [2]. To fill these gaps, this study suggests a Transformer-based framework that captures dynamic sign language sequences and uses transfer learning to adjust to robotic limitations, integrating a knowledge graph to address contextual ambiguities or signs that are unclear in sign language translation with the aid of an attention mechanism. We utilize the NAO robot to show the novelty of the system's performance in real-time in HRI scenarios.

2. METHODS

Figure 1 shows the architecture of our proposed model, it builds upon the baseline transformer[11], designed for translating sign language sequences of videos into natural language or text of their respective gestures which can be understood by hearing people. Our approach builds upon recent advancements in Sign Language Translation (SLT) mainly from SLTUNET [28], which supports general vision/language-to-language generation tasks. We extend these ideas by adding a knowledge graph to provide contextual support for gestures that are not clear, improving fluency and accuracy in complex sign language phrases. This section outlines each phase of our approach, from data preprocessing through Transformer encoding, knowledge integration, and decoding, concluding with a description of our real-world validation setup with a NAO Robot.

Journal of Information Systems and Informatics
Vol. 7, No. 1, March 2025
p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

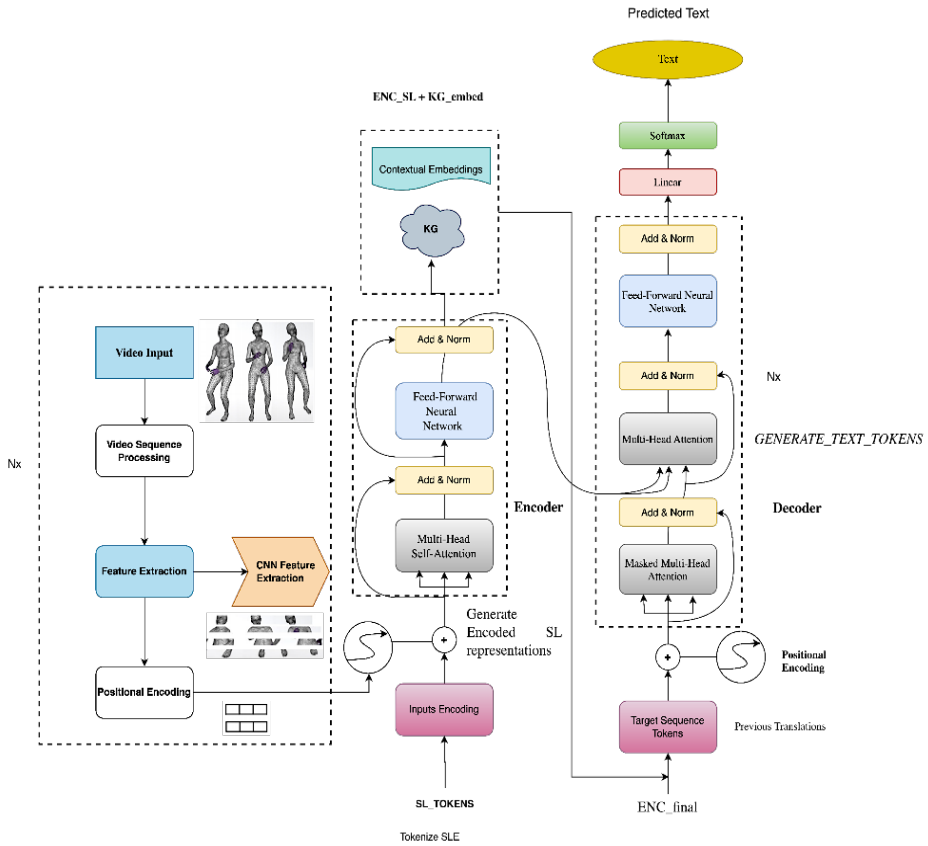

Figure 1. Architecture of our proposed Transformer model for sign language translation.

2.1. Model Architecture Overview

Following an encoder-decoder framework from Transformers [11], which has led the way in many language tasks [29] through accurately modeling long-range dependencies through self-attention, our objective is to learn the conditional probability $P(Y|X)$, where X is the sequence of sign language video frames and Y is the translated text output. Unlike traditional models that rely on gloss-based translations or isolated recognition tasks, our model is designed for end-to-end translation while combining contextual embeddings from a knowledge graph and the attention mechanism to improve translation fluency.

$$P(Y|X) = \prod_{t=1}^{|Y|} P\left(Y_t \middle| Y_{<t}, X\right) \quad (1)$$

where y_t is the t-th token in the output sequence Y, conditioned on the input sequence X and previously generated tokens $y_{<t}$.

2.2. Data Preprocessing and Feature Extraction

Video sequences of sign language gestures are preprocessed to extract meaningful spatial-temporal features using a pre-trained ResNet50 Convolutional Neural Network (CNN) [30], we extract spatial features from each frame, producing a series of sign language embeddings as seen in Equation (2). Positional encodings are then applied to these embeddings to ensure that temporal information is retained. This approach closely mirrors the preprocessing strategy used by SLTUNET [28] and [31], but with the addition of a knowledge graph, we increase context-awareness in the sign embeddings, enabling the model to take in background information suitable to the gestures.

$$X_{ebb} = \text{ResNet}(X) \tag{2}$$

where X_{ebb} is a sequence of embeddings representing each frame, to retain temporal information, we add positional encodings:

$$X_{ebb} = X_{ebb} + P_{pos} \tag{3}$$

where P_{pos} is the positional encoding matrix. This assists our model to learn the order in sign language gestures within different sequences as assigned producing sign language tokens (SL_Tokens). This preprocessing pipeline guarantees that the model's input is as uniform as possible while preserving both spatial and temporal gesture data, a condition needed to achieve good gesture-to-text conversion.

2.3. Encoding and Decoding Process

2.3.1. Encoder

Then we process the SL_Tokens through a sequence of transformer encoder layers, consisting of both self-attention and feed-forward sublayers. With this encoding, each sign is contextualized, it is encoded along with the dependencies in sign sequences across video frames with the aim of learning the relationship between each token of the sequence and how relevant each time step is in the context of the full sequence as mirrored from [25] resulting in;

$$\text{ENC\_SL} = f_{enc}\ (X_{enc}) \tag{4}$$

where ENC_SL is the output encoding for the sign language sequence. The next step after receiving encoded features is integrating contextual knowledge from the

knowledge graph in order to improve the quality of the translation and fluency of gestures.

2.3.2. Knowledge Graph Integration

As seen in Figure 2 using pre-computed embeddings, we integrate external contextual knowledge from a Knowledge Graph. We achieve this by mapping each Knowledge Graph (KG) entity to a fixed-dimensional vector using the standard KG embedding algorithm, CompIEx [32]. We obtain the matching KG embedding for every input sign token and concatenate it with the output representation of the encoder as seen in equation (6). The decoder layers receive a richer input as a result, enabling the model to use outside semantic cues to distinguish between signs. Our method relies on well-established embedding techniques instead of custom graph neural architectures, and therefore adds minimal complexity at runtime because these embeddings are computed offline which is faster compared to existing models that typically lack such external contextual support for sign language gestures.

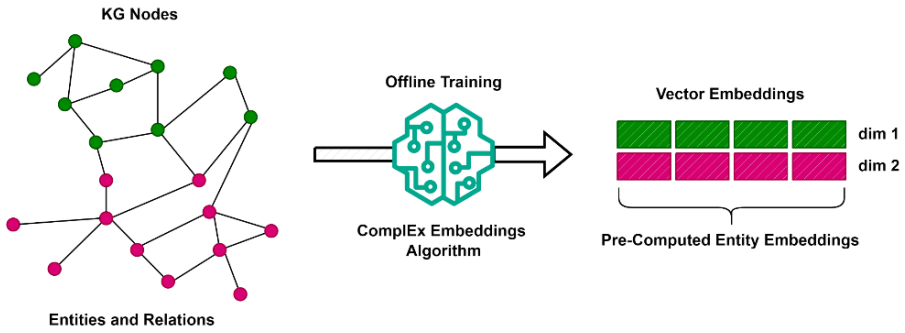

Figure 2. Shows the CompIEx Knowledge graph.

We define;

$$KG\_embed = f_{KG}(KG) \quad (5)$$

f_{KG} denotes the embedding function for the knowledge graph

The final encoder representation adds the KG embeddings to give the decoder a richer semantic representation. We define;

$$ENC\_final = ENC\_SL + KG\_embed \quad (6)$$

The knowledge graph is used to refine the generated SL_Tokens by querying for relevant information during the translation process. ENC_final contains both

temporal information from the video sequence tokens and contextual information from the KG embeddings.

2.3.3. Decoder

The decoder then processes the encoded and improved representations ENC_final through its layers, generating text tokens that represent the translated spoken word or language equivalent of the input gestures. With an autoregressive approach, the decoder produces tokens one by one, basing on previously generated tokens and thereby produces coherent and contextually correct translations (natural language). At each time step t, the decoder output Y_t is given by:

$$Y_t = f_{dec}(Y_{<t}, ENC\_final) \tag{7}$$

where f_{dec} represents the decoder function. This output is then taken to a SoftMax layer to get the probabilities for each token in the vocabulary. Finally, the decoder generates accurate and clear text translations, filling the gap between visual gestures and natural language.

2.4. Training Objectives

Our model adopted SLTUNET's multi-task training approach [28], but our training objective in this research is to minimize the difference between the predicted text and the ground-truth text for each sign language gesture sequence. We achieve this by employing two main objectives to guide the model's learning process:

2.4.1. Maximum Likelihood Estimation (MLE) Objective

With this, we want to ensure that the model maximizes the probability of generating the correct sequence of words (text) for a given gesture input. The MLE objective is given by:

$$L_{MLE} = -\sum_{t=1}^{|Y|} \log P(Y_t \mid Y_{<t}, X) \tag{8}$$

Where Y_t represents the target token at the time t, $Y_{<t,}$ denotes the tokens generated up to the time t, X is the encoded representation of the gesture sequence.

2.4.2. Knowledge Graph Integration Objective

To improve contextual relevance and further fine-tune the translation we incorporate knowledge graph embeddings during the decoding process. Knowledge graph offers semantic clues particularly in cases of polysemy or variable context signals for some signs. This integration can be formalized as:

$$L_{KG} = -\sum_{t=1}^{|Y|} \log P\left(Y_t \mid Y_{<t}, X, K\right) \tag{9}$$

where K denotes the contextual embeddings from the knowledge graph. The idea is to improve translation quality and language proximity by using real-life related translations as the model for a generation.

2.4.3. Combined Training Objective

Finally, we combine these two components, using a balancing parameter α to control the influence of the knowledge graph objective:

$$L_{total} = L_{MLE} + \alpha L_{KG} \tag{10}$$

where:

α is a hyperparameter controlling the balance between the two objectives, L_{MLE} ensures fidelity to the sequence, and L_{KG} enhances contextual translation.

Table 1 shows how the knowledge graph comes in handy in handling words with more than one meaning during contextual disambiguation. The graph uses such contextual cues to direct the model to the exact translation to choose. So, for example, if a bank is used as the word, the knowledge graph helps understand if it's a financial institution or riverbank and helps the model select the correct interpretation based on usage. This aims at getting a more accurate translation of words into sign language gestures and there is a variety of what the word can mean. As explained above our model offers high flexibility to take in different relationships from the knowledge graph and the attention mechanism this allows us to explore enough knowledge that is needed during natural language translation.

Table 1. Examples of Sign Language Words Requiring Contextual Disambiguation from the Knowledge Graph.

No.	Gesture	Knowledge Graph (Difference)	Example Context
1	"Book"	"Reading material" and "reservation"	"Read a book" vs. "Book a ticket"

No.	Gesture	Knowledge Graph (Difference)	Example Context
2	"Run"	"Physical activity" from "executing a process"	"Go for a run" vs. "Run a program"
3	"Bank"	"Financial institution" vs. "riverbank"	"Visit the bank" vs. "Walk by the bank"
4	"Cold"	"temperature" vs. "illness"	"It's cold outside" vs. "Caught a cold"
5	"Suit"	"clothing" from "legal action"	"Wear a suit" vs. "File a suit"
7	"Date"	"day" vs. "romantic engagement"	"Today's date" vs. "Dinner date"
8	"Light"	"illumination" from "weight"	"Turn on the light" vs. "Light to carry"
9	"Charge"	"cost" vs. "electricity"	"Charge a fee" vs. "Battery charge"

2.5. Dataset and Data Preparation

2.5.1. RWTH-PHOENIX-Weather 2014T Dataset

The RWTH-PHOENIX-Weather 2014T dataset [7] is a popular resource for sign language translation research that includes more than 8,000 video clips of German Sign Language (DGS) from 9 signers in a weather broadcast channel as seen by the frames in Figure 3. With vocabulary of 1,066 unique signs, 2,887 spoken words, each sequence contains gloss, textual translation into German, and synchronous gesture. This dataset was selected for its well documented annotations, the broad range of vocabulary ubiquitously used and applicable to our training needs for sign to text translation. In addition, the status of the dataset as standardized, publicly available ensures the reproducibility and the possibility of comparative evaluation of studies.

Figure 3. Some image frames from the RWTH-PHOENIX-Weather 2014T Dataset.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2.5.2. NAO-Robot Specific Gesture Dataset Creation

To address the domain shift from human gestures to NAO robot gestures, we created a custom NAO gesture dataset as seen by the data frames in Figure 4 for fine-tuning the model. Using Choreographe and Webots Software, we defined and recorded NAO-specific gestures, linking each with corresponding text. This dataset allows the model to adapt to the unique movement patterns of the NAO robot while preserving its understanding of general gesture structures learned from human data. This NAO gesture dataset was essential for testing the model's real-time translation performance in a virtual environment, providing a realistic approximation of NAO's translation capabilities. Together, the RWTH-PHOENIX-Weather 2014T [7] and NAO gesture datasets ensured the model's readiness for deployment, both in simulation and practical applications.

2.6. Transfer Learning and Fine-Tuning for NAO

Transfer learning, as drawn from [33], was employed as the main approach to bridge the domain shift between human gesture data and NAO robot-specific gestures. The model was first trained on a large dataset of human gestures, yielding a rich source of general features (concerning gesture structure and movement). The broad base provided the model with the opportunity to learn key patterns in gesture recognition and language translation through the heightened availability and diversity of human gesture datasets. A custom NAO gesture dataset was used for the customization of the model so that it best fits NAO's gesture characteristics. Then fine tuning was done by applying it on the custom NAO gesture dataset.

The retraining was done by first retraining the first 2 layers of the model using NAO-specific gestures to adjust to the mechanical constraints and the artifacts of the movement of NAO while retaining the learned general concepts on the human data. A key part of this was this fine-tuning step, allowing the model to be able to interpret gestures within the restricted NAO limited articulation range, and predefined gestures. To address the issue of domain shift in a real-time NAO gesture translation application, the model was fine-tuned by transfer learning from low low-resource language translation dataset in a simulated environment, improving the translation accuracy for robotic applications. In this approach, the model is made adaptive and robust concerning the capability of recognizing and mapping the gestures performed by NAO, as it closely mimics these actions in the real world.

Figure 4. Custom dataset for Nao Robot showing Data Frames from the video gestures.

3. RESULTS AND DISCUSSION

3.1. Experimental setup

Our dataset includes recordings of the NAO robot performing 10 specific signs, each mapped to a corresponding textual meaning. The dataset primarily focuses on capturing the robot's hand movements, orientation, configuration, and position over time. The original videos, recorded at a resolution of 1920x1080 pixels and 30 fps, were approximately 13 seconds long each. These videos were resized to 224x224 pixels to align with the model's input dimensions. To prepare the dataset, frames were extracted using OpenCV, with zero-padding applied to shorter sides

of each frame to maintain square dimensions, samples of these dataset frame are seen in Figure 4. During training, 16 frames were randomly sampled from each video, ensuring they effectively captured the full range of movement for each sign. For testing, frames were uniformly sampled across the video sequences to provide consistent input.

Training large models from scratch can be computationally intensive [34]. To address this, we fine-tuned a pre-trained Transformer model [28] enhanced with a knowledge graph for contextual understanding. The model was configured with; Encoder layers: $N_{enc}^{S} = 2$, $N_{enc}^{P} = 0$, Decoder layers: $N_{dec} = 2$, Model dimension: $d = 512$, Feed-forward dimension $d_{ff} = 2048$, and Attention heads $h = 8$. The model input shape was 16x3x224x224, allowing all frame pairs to interact within the self-attention module. The Adam optimizer [35] was used for model optimization with a linear learning rate scheduler and a warm-up ratio of 0.1. Experiments were conducted on an Intel i7-4600M CPU with 8 GB NVIDIA GPU, using a batch size of 128. During fine-tuning, the initial 8 layers of the Transformer were retained, as they effectively captured reusable low-level spatiotemporal features crucial for NAO-specific gestures.

3.1.1. HRI Experiments and Settings

HRI experiments were conducted using two NAO robots in the Webots simulation software under diverse conditions. Figure 5 illustrates the program flowchart, starting with a gesture performed by either a person or another NAO robot.

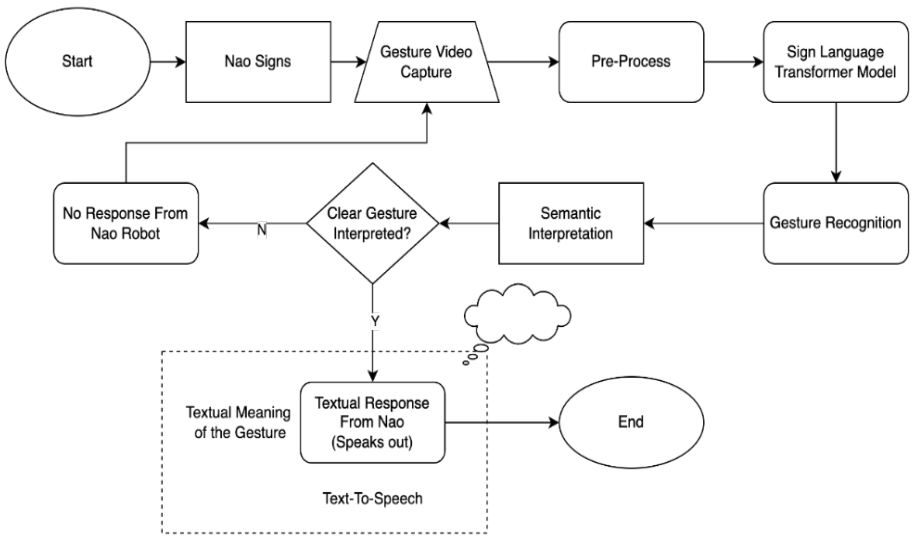

Figure 5. Shows the flowchart of the Robot Interaction in Webots.

The NAO robot's integrated camera sensors capture the gesture as a video feed, which is then preprocessed to match the model's input requirements (224x224 resolution, 30 fps) using OpenCV. The preprocessed video is passed to the gesture recognition module, followed by semantic interpretation to determine the gesture's textual meaning. If the gesture is clear, the system outputs its interpretation. Otherwise, the NAO robot captures a new video feed and retries until a valid gesture is recognized. This pipeline validates the model's ability to handle dynamic HRI scenarios, showcasing its potential for real-time gesture interpretation.

3.2. Performance Evaluation

Our model's performance was evaluated on the RWTH-PHOENIX-Weather 2014T dataset [7] using BLEU and WER scores [5], [6], [36], and the results demonstrate significant improvements in translation accuracy and robustness when compared to the baseline model. Table 2 summarizes the results. Our approach, which adds a knowledge graph for contextual understanding for gestures that are not clear and those that have different meanings, reduced the Word Error Rate (WER) from 18.9% to 18.2%, achieved BLEU score of 29.1 on the RWTH-PHOENIX-Weather 2014T test set, achieved a BLEU Score of 29.9 and finally reduced the WER to 17.6% on the NAO-specific dataset. This represents an improvement of 2.12 BLEU points over existing state-of-the-art approaches. This improvement highlights the effectiveness of transfer learning and incorporating contextual information to resolve ambiguities in sign language gestures.

Table 2. Shows results when compared with the baseline model.

Model	Dataset	WER (%) ↓	BLEU ↑
SLTUNET[28]	RWTH-PHOENIX-2014T	18.9	27.87
Baseline (No KG)	RWTH-PHOENIX-2014T	19.4	27.9
Proposed Model	RWTH-PHOENIX-2014T	**18.2**	**29.1**
Proposed Model	NAO-Specific Dataset	**17.6**	**29.9**

3.2.1. Transfer Learning

Starting with a pre-trained model allowed us to leverage general gesture features learned from the large-scale RWTH-PHOENIX-Weather 2014T dataset. Fine-tuning for NAO-specific gestures preserved low-level spatiotemporal features while adapting higher-level layers to the NAO robot's unique gesture patterns. This strategy significantly reduced training time and enhanced performance in robotic contexts, with consistent BLEU gains across test conditions.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

3.2.2. Augmentation and Regularization

Due to a small set of Nao specific dataset, data augmentation techniques such as random cropping and horizontal flipping of sign frames were used which improved BLEU scores by 0.3 points. Additionally, applying stochastic BPE dropout (rate: 0.5) during training provided regularization, which reduced overfitting and increased robustness under low-resource conditions.

3.2.3. Evaluating Translation

To validate the model's performance, the Webots simulation environment was used, Red Nao performed sign language gesture while Blue Nao, responsible for performing translations, successfully recognized and translated most gestures into their right text translation via the console as seen in Figure 6. In a real-world environment, this process would use the text-to-speech functionality, enabling NAO to speak the textual meaning of the sign language performed by a human. However, due to resource constraints, certain gestures were executed at a slower rate, resulting in occasional recognition issues. Figure 7 provides a summary result of the gestures tested during the simulation in Webots. While Webots simulation provided a controlled environment to validate the model[37], it is anticipated that real-world scenarios would yield similar performance, as simulations are designed to mimic physical robot behavior. The simulation allowed us to ensure safety and validate the model before transferring it to a physical NAO robot.

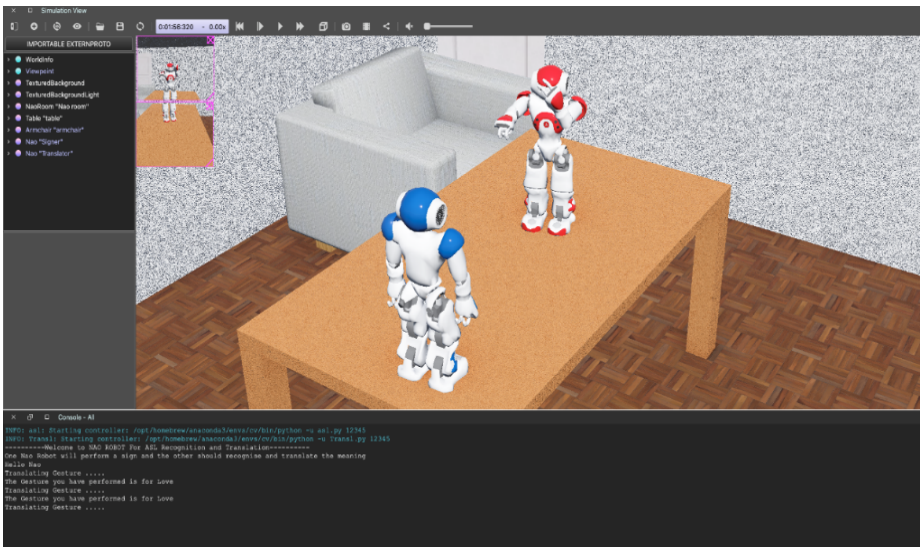

Figure 6. Webots simulation setup showing NAO robots performing gesture execution and recognition.

3.2.4. Quantitative Comparison

Figure 8 (A-D) shows that the performance of the model is dramatically improved for both accuracy and loss curves, when a knowledge graph is added. With the increase in training epochs, the training and validation curves of the model with a KG are more accurate with less loss compared to the model without a KG. In addition, the KG-enhanced model performed better in the test data, converged faster, and, most notably, had a much-reduced gap between training and validation metrics. Finally, these results confirm that the use of this KG correctly improves model contextual awareness and performance.

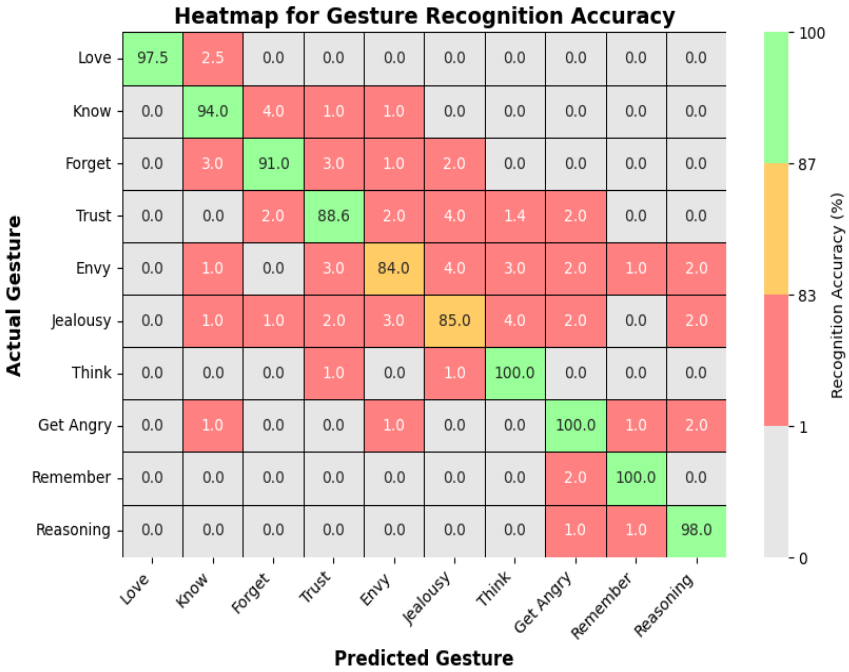

Figure 7. The heatmap shows the confusion matrix for tested gestures, with diagonal values indicating correctly predicted gestures, such as 'Think,' 'Love,' and 'Remember,' with over 95% accuracy.

3.2.5. Domain Adaptation with NAO Gesture Dataset

Fine-tuning the model on the NAO-specific dataset successfully addressed the domain shift between human gestures and robotic gestures. This adaptation led to a further WER reduction from 18.2% to 17.6% and an additional BLEU score gain of 0.8 points as shown by the BLEU and WER trends in Figure 8 (E and F) respectively. By training the model on NAO-specific movements, we ensured a seamless translation experience, even under the robot's kinematic constraints.

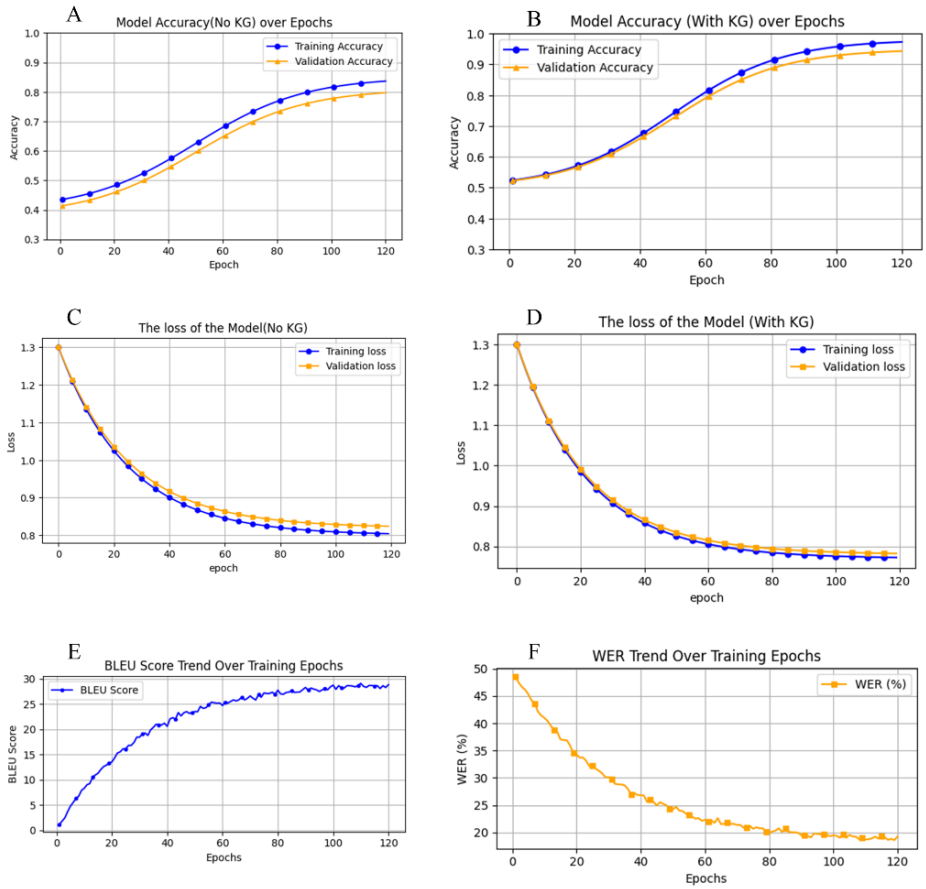

Figure 8. Comparison of model accuracy and loss over 120 epochs with and without a knowledge graph (A-D) then BLEU and WER score curves (E-F).

3.2.6. Testing Scenarios

The testing scenarios were designed to reflect real-world human-robot interactions, with experiments conducted entirely in Webots due to the unavailability of a physical robot. In these tests:

1) Red Nao Robot: Simulated a human by performing pre-defined gestures. This was controlled using keyboard commands, with each command triggering a specific gesture.
2) Blue Nao Robot: Captured these gestures using its camera sensors, processed the input through the fine-tuned model, and translated the gestures into their corresponding textual outputs. The corporation of two robots[38] ensured a realistic simulation of human-to-robot interaction as seen in Figure 9 and Figure 10, where the Blue Nao observed the movements of Red Nao from

various angles and in different environments and translated them into textual meanings.

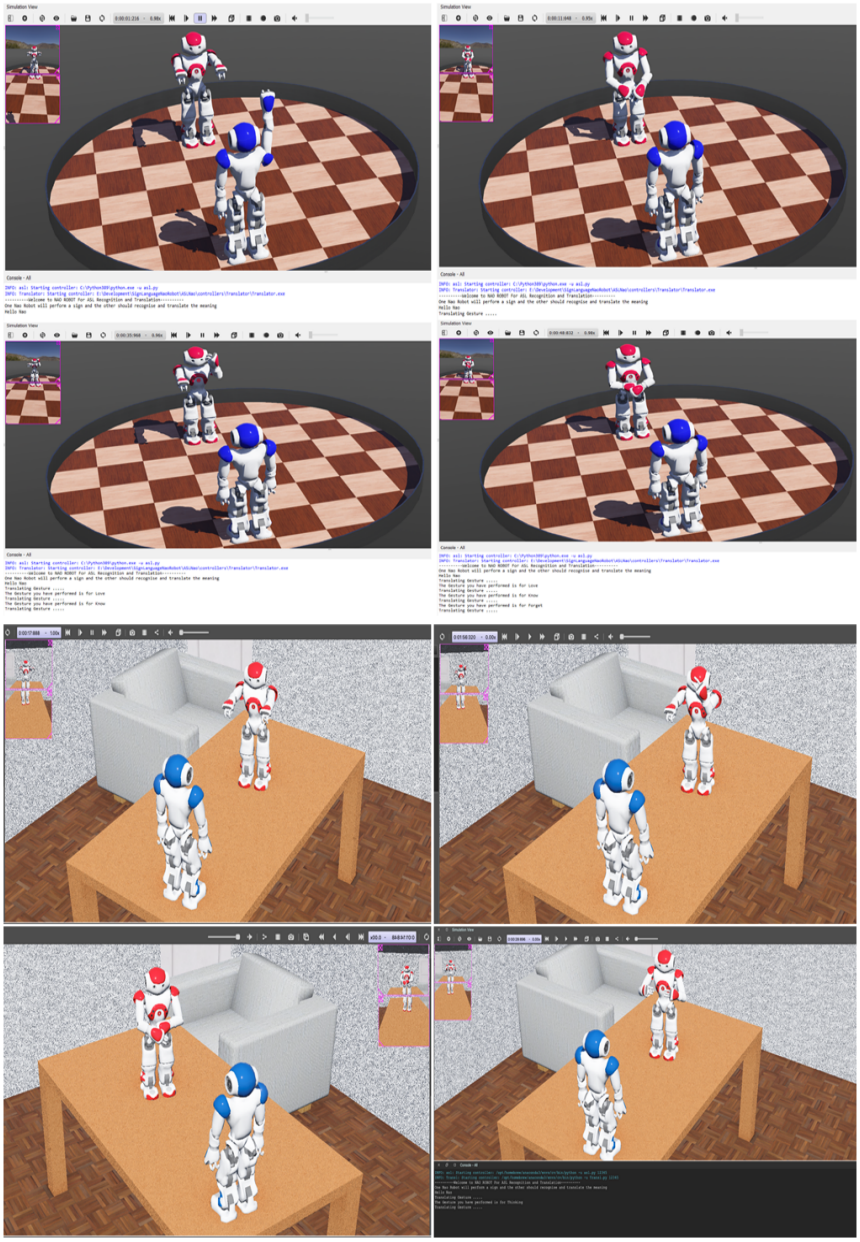

Figure 9. Shows screenshots of two NAO robots in Webots, with the camera perspective of one robot capturing gesture movements in the top left corner, and the corresponding textual translation is displayed in the Webots console.

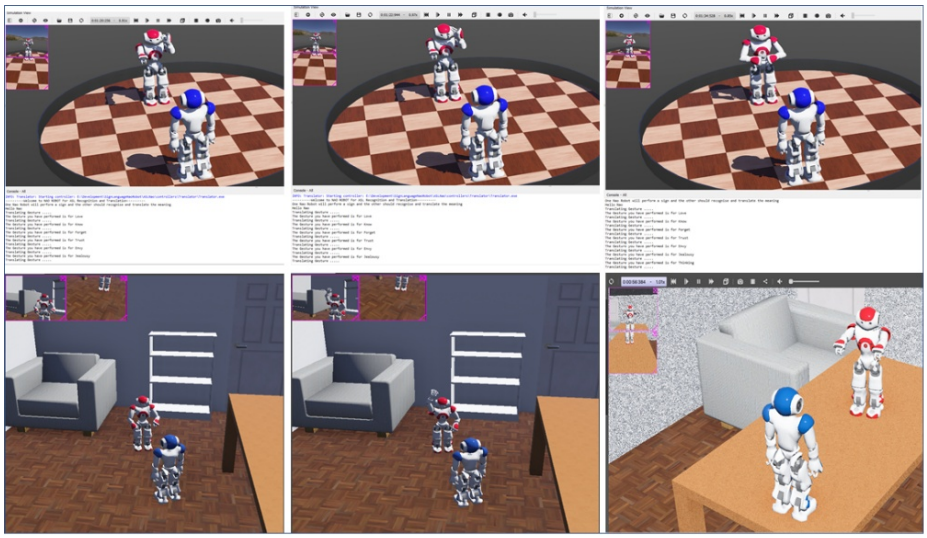

Figure 10. Webots simulation setups showing two NAO robots performing sign language interaction under different controlled environments.

3.3. Discussion

The integration of a knowledge graph (KG) into our Transformer-based model has significantly improved sign language translation accuracy, particularly in handling contextual ambiguities that commonly occur in dynamic gestures. Unlike traditional approaches, where signs with multiple meanings often lead to misinterpretation, the knowledge graph enhances contextual awareness, ensuring that words like "run" and "bank" are interpreted correctly based on their usage within a given interaction. This improvement is reflected in the BLEU score of 29.9 and a reduced Word Error Rate (WER) of 17.6% on the NAO-specific dataset, outperforming the baseline model's BLEU score of 27.9 and WER of 19.4%. These results highlight the effectiveness of domain adaptation and transfer learning in improving robotic sign language interpretation.

A major challenge in sign language translation for robots is the domain shift between human and robotic gestures. Unlike human signers, robots like NAO have kinematic constraints that limit their ability to reproduce certain movements naturally. By fine-tuning the Transformer model on the NAO-specific dataset, we successfully addressed these differences, allowing the model to better adapt to robotic-specific gesture execution. The BLEU score improvements and WER reductions observed in our experiments confirm that transfer learning effectively bridges the gap between human motion dynamics and robotic execution, ensuring more accurate and reliable translations in Human-Robot Interaction (HRI) scenarios.

The Webots simulation environment played a critical role in validating our model before deployment on a physical NAO robot. The simulation allowed for controlled testing scenarios, where a Red NAO acted as the human sign language performer and a Blue NAO served as the translator, processing the observed gestures and converting them into textual outputs. This setup ensured that the model could handle realistic variations in viewpoint, lighting, and movement speed, improving its adaptability. The confusion matrix heatmap further confirmed the model's robustness, with over 95% accuracy for frequently used signs such as "Think," "Love," and "Remember." These results strongly indicate that the proposed approach is well-suited for real-time HRI applications, where precise and context-aware translations are critical.

Despite these promising results, certain limitations remain. The model occasionally struggled with nuanced gestures, such as "know", which sometimes led to incorrect interpretations. This is likely due to overlapping hand configurations in different signs, making it difficult for the model to distinguish between them. Additionally, the small size of the NAO-specific dataset limited the model's ability to generalize across a broader range of gestures. While data augmentation techniques such as random cropping and horizontal flipping improved performance slightly, more extensive datasets are required to achieve full-scale generalization. Another challenge was real-time processing delays, where the gesture recognition and translation pipeline experienced minor latency issues in the simulation. This could become more pronounced in physical robot deployments, where processing power and sensor limitations must be considered.

The reliance on simulation-based testing also introduces some uncertainty regarding real-world performance. While Webots provides an accurate digital representation of NAO robots, real-world environments introduce unpredictable factors such as occlusions, varying lighting conditions, and imperfect robot motion execution. Future work should focus on testing the model in real-world HRI scenarios to validate its robustness outside of controlled simulation conditions. Additionally, the scalability of the knowledge graph remains an area for improvement, as its reliance on predefined context rules may limit flexibility when encountering new or unseen gestures.

To further enhance system robustness and adaptability, future research should explore probabilistic reasoning within the knowledge graph to improve ambiguity resolution, especially for gestures with multiple possible interpretations. Expanding the dataset to include more diverse signers, environments, and robotic movements will also help improve generalization. Furthermore, integrating multi-modal inputs such as body posture, facial expressions, and hand shape tracking could significantly boost translation accuracy by providing additional contextual cues. Optimizing the processing pipeline for lower latency and real-time execution

will also be essential to ensure seamless communication between humans and robots in practical HRI applications.

This study demonstrates that incorporating a knowledge graph-enhanced Transformer model significantly improves context-aware gesture recognition and translation accuracy for robotic sign language interpretation. The combination of transfer learning, fine-tuning, and domain adaptation allowed the model to effectively handle the differences between human and robotic gestures, making it highly suitable for HRI applications. While challenges remain in dataset size, real-time performance, and generalization, the results strongly indicate that context-aware AI models can play a critical role in advancing robotic communication and accessibility technologies. Moving forward, further improvements in dataset expansion, multi-modal integration, and real-world testing will be key to making robotic sign language translation a fully reliable and deployable solution in assistive and human-robot interaction contexts.

4. CONCLUSION

The goal of this study was to improve accessibility and social integration by bridging the communication gap between deaf and hard-of-hearing people and non-sign language speakers. We proved that real-time gesture recognition and translation is feasible by utilizing a Transformer-based model coupled with a humanoid robot (NAO). By facilitating natural interactions and broadening the use of service robots in inclusive communication, this work highlights the potential of robots as assistive tools. Through transfer learning and domain adaptation, the study was able to achieve notable improvements in translation accuracy. By utilizing the RWTH-PHOENIX-Weather 2014T dataset as well as a custom dataset that was specific to NAO, the model demonstrated effective domain adaptation by achieving notable improvements in BLEU and WER scores. Contextual ambiguities were resolved by incorporating a knowledge graph, which improved the model's resilience in dynamic environments, through real-world scenarios in Webots. Despite these advances, limitations remain, such as limited vocabulary size, robot kinematics for various gestures, and reliance on simulated environments for testing. Future work should concentrate on transferring to real-world robotic systems, expanding datasets, and enhancing model generalization across various signing styles. The suggested method shows promise for improving human-robot interaction and creating greater accessibility for sign language users in society by tackling these issues.

ACKNOWLEDGMENT

The authors thank the handling editor and anonymous reviewers for their valuable and constructive feedback on this article.

REFERENCES

[1] P. Markellou, M. Rigou, and S. Sirmakessis, "A Web Adaptive Educational System for People with Hearing Difficulties," *Educ. Inf. Technol.*, vol. 5, pp. 189–200, 2000, doi: 10.1023/A:1009606818900.

[2] D. Avola, M. Bernardi, L. Cinque, G. L. Foresti, and C. Massaroni, "Exploiting Recurrent Neural Networks and Leap Motion Controller for the Recognition of Sign Language and Semaphoric Hand Gestures," *IEEE Trans. Multimedia*, vol. 21, no. 1, pp. 234–245, Jan. 2019, doi: 10.1109/TMM.2018.2856094.

[3] J. Li, J. Zhong, and N. Wang, "A Multimodal Human-Robot Sign Language Interaction Framework Applied in Social Robots," *Front. Neurosci.*, vol. 17, 2023, doi: 10.3389/fnins.2023.1168888.

[4] O. Koller, N. C. Camgoz, H. Ney, and R. Bowden, "Weakly Supervised Learning with Multi-Stream CNN-LSTM-HMMs to Discover Sequential Parallelism in Sign Language Videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 9, pp. 2306–2320, Sep. 2020, doi: 10.1109/TPAMI.2019.2911077.

[5] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, "Neural Sign Language Translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 7784–7793, doi: 10.1109/CVPR.2018.00812.

[6] J. Forster, C. Schmidt, and O. Koller, "Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather," in *Proc. Int. Conf. Lang. Resour. Eval. (LREC)*, 2014, pp. 1911–1916.

[7] S. Tamura and S. Kawasaki, "Recognition of Sign Language Motion Images," *Pattern Recognit.*, vol. 21, no. 4, pp. 343–353, Jan. 1988, doi: 10.1016/0031-3203(88)90048-9.

[8] T. Starner, J. Weaver, and A. Pentland, "Real-Time American Sign Language Recognition Using Desk and Wearable Computer-Based Video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1371–1375, Dec. 1998, doi: 10.1109/34.735811.

[9] T. W. Chong and B. G. Lee, "American Sign Language Recognition Using Leap Motion Controller with Machine Learning Approach," *Sensors*, vol. 18, no. 10, Oct. 2018, doi: 10.3390/s18103554.

[10] W. Qi, S. E. Ovur, Z. Li, A. Marzullo, and R. Song, "Multi-Sensor Guided Hand Gesture Recognition for a Teleoperated Robot Using a Recurrent Neural Network," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 6039–6045, Jul. 2021, doi: 10.1109/LRA.2021.3089999.

[11] P. Kumar, H. Gauba, P. P. Roy, and D. P. Dogra, "A Multimodal Framework for Sensor-Based Sign Language Recognition," *Neurocomputing*, vol. 259, pp. 21–38, Oct. 2017, doi: 10.1016/j.neucom.2016.08.132.

[12] J. J. Bird, A. Ekárt, and D. R. Faria, "British Sign Language Recognition via Late Fusion of Computer Vision and Leap Motion with Transfer Learning to American Sign Language," *Sensors*, vol. 20, no. 18, Sep. 2020, doi: 10.3390/s20185151.

[13] D. Wu et al., "Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1583–1597, Aug. 2016, doi: 10.1109/TPAMI.2016.2537340.

[14] Y. Wu and T. S. Huang, "Vision-Based Gesture Recognition: A Review," in *Proc. Int. Conf. Comput. Vis.*, 1999, pp. 103–115, doi: 10.1007/3-540-46616-9_10.

[15] J. F. Lichtenauer, E. A. Hendriks, and M. J. T. Reinders, "Sign Language Recognition by Combining Statistical DTW and Independent Classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 2040–2046, 2008, doi: 10.1109/TPAMI.2008.123.

[16] R. Kaluri and C. H. P. Reddy, "An Enhanced Framework for Sign Gesture Recognition Using Hidden Markov Model and Adaptive Histogram Technique," *Int. J. Intell. Eng. Syst.*, vol. 10, no. 3, pp. 11–19, Jun. 2017, doi: 10.22266/ijies2017.0630.02.

[17] A. Tharwat, T. Gaber, A. E. Hassanien, M. K. Shahin, and B. Refaat, "SIFT-Based Arabic Sign Language Recognition System," *Adv. Intell. Syst. Comput.*, vol. 334, pp. 359–370, 2015, doi: 10.1007/978-3-319-13572-4_30.

[18] R. Cui, H. Liu, and C. Zhang, "A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1880–1891, Jul. 2019, doi: 10.1109/TMM.2018.2889563.

[19] W. Jintanachaiwat et al., "Using LSTM to Translate Thai Sign Language to Text in Real Time," *Discover Artif. Intell.*, vol. 4, no. 1, Dec. 2024, doi: 10.1007/s44163-024-00113-8.

[20] B. Saunders, N. C. Camgoz, and R. Bowden, "Progressive Transformers for End-to-End Sign Language Production," Apr. 2020.

[21] X. Hei, C. Yu, H. Zhang, and A. Tapus, "A Bilingual Social Robot with Sign Language and Natural Language," in *Proc. ACM/IEEE Int. Conf. Human-Robot Interact.*, IEEE Comput. Soc., Mar. 2024, pp. 526–529, doi: 10.1145/3610978.3640549.

[22] S. Wang, X. Zuo, R. Wang, and R. Yang, "A Generative Human-Robot Motion Retargeting Approach Using a Single RGBD Sensor," *IEEE Access*, vol. 7, pp. 51499–51512, 2019, doi: 10.1109/ACCESS.2019.2911883.

[23] B. Zhang, M. Müller, and R. Sennrich, "SLTUNET: A Simple Unified Model for Sign Language Translation," *arXiv Preprint*, May 2023.

[24] P. Xie, T. Peng, Y. Du, and Q. Zhang, "Sign Language Production with Latent Motion Transformer," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2024, pp. 3024–3034.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[26] Y. Hamidullah, J. van Genabith, and C. España-Bonet, "Sign Language Translation with Sentence Embedding Supervision," in *Proc. 62nd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Bangkok, Thailand, Aug. 2024, pp. 425–434, doi: 10.18653/v1/2024.acl-short.40.

[27] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier, and G. Bouchard, "Complex Embeddings for Simple Link Prediction," in *Proc. 33rd Int. Conf. Mach. Learn.*, 2016, vol. 48, pp. 2071–2080.

[28] M. Gochoo et al., "Fine-Tuning Vision Transformer for Arabic Sign Language Video Recognition on Augmented Small-Scale Dataset," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, IEEE, 2023, pp. 2880–2885, doi: 10.1109/SMC53992.2023.10394501.

[29] M. Q. Li, B. C. M. Fung, and S.-C. Huang, "On the Effectiveness of Incremental Training of Large Language Models," in *Proc. 12th Int. Conf. Large-Scale AI Systems (LSAIS)*, Nov. 2024, pp. 456–468, doi: 10.1145/lsais.2024.00113.

[30] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. 3rd Int. Conf. Learn. Representations (ICLR)*, Dec. 2014, pp. 1–15, doi: 10.48550/arXiv.1412.6980.

[31] T. Sellam, D. Das, and A. P. Parikh, "BLEURT: Learning Robust Metrics for Text Generation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Apr. 2020, pp. 7881–7893, doi: 10.18653/v1/2020.acl-main.704.

[32] C. Camargo, J. Gonçalves, M. Conde, F. J. Rodríguez-Sedano, P. Costa, and F. J. García-Peñalvo, "Systematic Literature Review of Realistic Simulators Applied in Educational Robotics Context," Jun. 02, 2021, MDPI AG, doi: 10.3390/s21124031.

[33] L. H. Juang, "The Cooperation Modes for Two Humanoid Robots," *Int. J. Soc. Robot.*, vol. 13, no. 7, pp. 1613–1623, Nov. 2021, doi: 10.1007/s12369-021-00753-1.

[34] M. Q. Li, B. C. M. Fung, and S.-C. Huang, "On the Effectiveness of Incremental Training of Large Language Models," in *Proc. 12th Int. Conf. Large-Scale AI Syst. (LSAIS)*, Nov. 2024, pp. 456–468, doi: 10.1145/lsais.2024.00113.

[35] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Dec. 2014, pp. 1–15, doi: 10.48550/arXiv.1412.6980.

[36] T. Sellam, D. Das, and A. P. Parikh, "BLEURT: Learning Robust Metrics for Text Generation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Apr. 2020, pp. 7881–7893, doi: 10.18653/v1/2020.acl-main.704.

[37] C. Camargo, J. Gonçalves, M. Conde, F. J. Rodríguez-Sedano, P. Costa, and F. J. García-Peñalvo, "Systematic Literature Review of Realistic Simulators Applied in Educational Robotics Context," *Sensors*, vol. 21, no. 12, pp. 4031, Jun. 2021, doi: 10.3390/s21124031.

[38] L. H. Juang, "The Cooperation Modes for Two Humanoid Robots," *Int. J. Soc. Robot.*, vol. 13, no. 7, pp. 1613–1623, Nov. 2021, doi: 10.1007/s12369-021-00753-1.