SignLLM: Sign Languages Production Large Language Models

Sen Fang[1], Lei Wang[2,3], Ce Zheng[4]
Yapeng Tian[5], and Chen Chen[6]

[1]Rutgers University, [2]Australian National University, [3]Data61/CSIRO, [4]Carnegie Mellon University, [5]University of Texas at Dallas, [6]University of Central Florida
sen.fang@rutgers.edu

Abstract. In this paper, we introduce the first comprehensive multilingual sign language dataset named Prompt2Sign, which builds from public data including American Sign Language (ASL) and seven others. Our dataset transforms a vast array of videos into a streamlined, model-friendly format, optimized for training with translation models like seq2seq and text2text. Building on this new dataset, we propose SignLLM, the first multilingual Sign Language Production (SLP) model, which includes two novel multilingual SLP modes that allow for the generation of sign language gestures from input text or prompt. Both of the modes can use a new loss and a module based on reinforcement learning, which accelerates the training by enhancing the model's capability to autonomously sample high-quality data. We present benchmark results of SignLLM, which demonstrate that our model achieves state-of-the-art performance on SLP tasks across eight sign languages. More code and materials are available at https://signllm.github.io/.

Keywords: Sign Language Production · Large Language Model · Transformers · Sequence-to-Sequence · Human Pose Generation

1 Introduction

Sign Language Production (SLP) aims to synthesize human-like sign avatars from text inputs. Deep learning-based SLP approaches [37, 71–74] typically involves sequential steps from text to gloss (*i.e.*, a type of textual vocabulary representing gestures or postures), gloss to pose [9, 52], and finally rendering pose videos into more engaging human-like avatar videos. These processes are complex and challenging to simplify, making sign language data acquisition and processing difficult. This challenge has significantly dampened researchers' enthusiasm and progress over a considerable period, with the majority of studies in the past decade relying on a German sign language (GSL) dataset named PHOENIX14T [25, 45] and other lesser-known languages datasets [20, 26, 32, 58, 68, 81] for **S**ign **L**anguage **P**roduction, **R**ecognition, and **T**ranslation tasks (SLP, SLR and SLT). Additionally, the use of various tools adhering to different standards [31,51,84] by different researchers further complicates the task. Recent research work [3,23,87] based on the American sign language (ASL) dataset [21, 79] is almost nascent, and the research progress on other minority languages is also very limited [22,42].

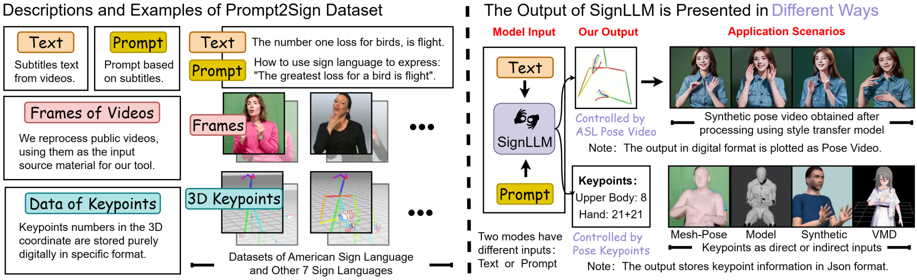

Fig. 1: Overview:(left) Major components (*e.g.*, Text, Prompt, Data of Keypoints, *etc.*) of our Prompt2Sign dataset. Data of Keypoints is the posture data that has been reprocessed, which is more useful data that is suitable for training. The input of our reprocessing tool is public videos from the datasets or the internet; (right) Our proposed SignLLM aims to generate sign language poses for various application scenarios [5, 11, 105, 110].

The existing mainstream datasets [13, 21] have played a significant role in advancing the field. However, they fall short in addressing the emerging challenges: (1) These existing datasets consist of complex format files including images, scripts, OpenPose [8] skeletal keypoints and graphs (or possibly other preprocessing formats [9, 51, 52, 63]). These formats lack actionable information that can be directly trained. For instance, training directly on images that contains excessive redundant information is difficult for video-level SLP. A more straightforward way is to distill the key gestures/postures into text for training. This limitation hampers the generalization and broad usability of language models. (2) Manual annotation for gloss is labor-intensive and time-consuming. (3) Some sign video datasets are obtained from sign language professionals, and then reprocessed into different data formats. As a result, scaling up the dataset becomes exceedingly challenging. These limitations collectively impede the development and training of more advanced models in sign language processing.

To solve these issues, we introduce **Prompt2Sign**, a new dataset focusing on upper body movements in large-scale sign language demonstrators. The dataset overview is shown in Figure 1 (left), showcasing prompts, video subtitles, and files containing digital keypoints information. To create this dataset, we first use OpenPose [8] for video processing to standardize pose information of video frames (Frames of Videos, the original material of tool) into our predefined format. Storing keypoints information in our standardized format (Data of Keypoints) can reduce redundancy and facilitate training with seq2seq and text2text models. Subsequently, we reduce reliance on manual annotations by auto-creating prompt words to improve cost-effectiveness. Finally, we improve the processing level of automation for the tools, making the tools highly efficient and lightweight, requiring no additional model loading to improve the ability to process data (*i.e.*, solving the difficulty in manual preprocessing and data collection challenges above). Our new dataset sourced from publicly available sign language datasets and videos on the internet, reducing the limitations of previous datasets and covering eight different sign languages (specific language can be seen later in Section 3), making it the first comprehensive multilingual sign language dataset.

Meanwhile, we recognize the existing model [10, 71, 73, 83, 96, 97] needs improvement because training models with our new datasets brings new challenges: (1) Different sign language data cannot normally be trained concurrently due to the differences in the sign language of different countries. (2) Handling more languages and larger datasets results in slow and challenging training processes, with difficulties in downloading, storing, and data loading. So it is necessary to explore high-speed training methods. (3) The existing model structure cannot grasp more languages and understand more complex, natural human conversational inputs (*i.e.*, how to enhance the generalization capability of the large model and its fundamental understanding prompts ability), it is necessary to explore the areas that have been overlooked by previous researchers, such as multilingual SLP, efficient training, and the ability of understanding prompts.

To overcome these challenges, we introduce **SignLLM**, the first large-scale multilingual Sign Language Production (SLP) model developed based on our Prompt2Sign dataset, that produces the sign language skeletal poses of eight languages from texts or prompts. Our SignLLM has two distinct modes: (i) Multi-Language Switching Framework (MLSF), which allows multiple sign languages production in parallel by dynamically adding encoder-decoder groups. (ii) Prompt2LangGloss, allowing SignLLM to support static single-set encoder-decoder generation. Figure 1 (right) shows our model inputs and outputs, text is the input of mode (i), and prompt is the input of mode (ii). Two modes deal with different use cases: The MLSF is to achieve efficient multilingual SLP without causing semantic confusion; The Prompt2LangGloss is a user-friendly multilingual SLP mode that aims to understand more complex natural language input. Our motivation is to establish a paradigm for multilingual generation and recognition based on our new dataset. To address the problem of extended training time caused by more languages and larger datasets, we present a new loss function with a unique module based on Reinforcement Learning (RL) concept to accelerate the training process of models on more languages, larger datasets.

We conducted extensive experiments and ablation studies. The results validate the superior performance of our **SignLLM** over baseline approaches [10, 22, 42, 71, 73, 83, 96, 97] on the dev and test sets in eight sign languages. We also provide qualitative evaluation, social impact, and limitations for discussion. The contributions of this paper can be summarised as:

- The first comprehensive multilingual sign language dataset, with more vocabulary words and eight languages, accommodates a greater variety of models. Its accompanying tools facilitate more automated/clean sign language data processing.
- The first large sign language production model, with two multilingual Sign Language Production (SLP) modes, achieves SOTA performance in SLP tasks across eight sign languages. The two modes prepare for accurately generating and understanding complex inputs in two use cases, respectively.
- A novel loss function, accompanied by a corresponding functional module, was introduced as a training strategy based on reinforcement learning in sign language firstly, aiming to reduce time costs in training.

Table 1: Comparison of Dataset Details: Prompt2Sign uses tools to automate the acquisition and processing of sign language videos on the web, is a better dataset that is efficient (a higher level of preprocessing, standardized and more models available), and lightweight (average reduction of 80% in space usage). Languages included: American Sign Language (ASL), German Sign Language (GSL, Alias DGS), Swiss German Sign Language (DSGS), French Sign Language of Switzerland (LSF-CH), Italian Sign Language of Switzerland (LIS-CH), Argentine Sign Language (Lengua de Señas Argentina, LSA), Korean Sign Language (KSL), and Turkish Sign Language (TSL).

Name	Language	Vocab.	Duration (h)	Signers	Multiview	Transcription	Gloss	Pose	Depth	Speech	Prompt	Compress
Video-Based CSL [36]	CSL	178	100	50	✗	✓	✗	✓	✓	✗	✗	✗
SIGNUM [93]	GSL	450	55	25	✗	✓	✓	✗	✗	✗	✗	✗
RWTH-Phoenix-2014T [13]	GSL	3k	11	9	✗	✓	✓	✗	✗	✗	✗	✗
Public DGS Corpus [33]	GSL	–	50	327	✓	✓	✓	✓	✗	✗	✗	✗
BSL Corpus [76]	BSL	5k	–	249	✗	✓	✓	✗	✗	✗	✗	✗
NCSLGR [59]	ASL	1.8k	5.3	4	✓	✓	✓	✗	✗	✗	✗	✗
How2Sign [21]	ASL	16k	79	11	✓	✓	✓	✓	✓	✓	✗	✗
Prompt2Sign (ours)	Multilingual	40k	200	40	✓	✓	✓	✓	✓	✓	✓	✓

2 Related Work

Sign Language Datasets. In recent years, the field of sign language research has primarily focused on Sign Language Recognition (SLR) [15,17,29,39,44,46] and Sign Language Translation (SLT) [3,6,14,43] based on deep learning. Traditional synthetic animation-based methods [16,41,55,56,77] have been gradually discarded. Due to a lack of suitable, high-quality datasets for ASL, deep learning researchers conducted their research [37,71–74] based on a GSL weather theme dataset, released in 2012 [25,45]. As previously mentioned, the data processing involved in sign language research is highly complex. Even with the release of the first large-scale ASL dataset[1] in 2021 [21], work focused on ASL-related themes based on it has not emerged quickly, as existing work is not easily transferable (the situation is worse for minority languages). We strive to standardize processing to address challenges related to data collection, utilization, and storage.

Large Language Models. LLMs refer to giant transformer models trained on extensive textual data, exhibit capabilities in understanding natural language and addressing complex tasks [4, 12, 78, 88, 90]. Sign language is a visual language, theoretically different from language models. However, most current work uses text2text and seq2seq models [48,61,66,100], converting key points/dense maps/grid poses into sequences for training, as opposed to directly training images. Hence, viewing the core process of SLP, text2pose, as a language model is justifiable. Extensive research indicates that an increase in parameters or data volume [34,40] significantly enhances the abilities of LLMs [4,12,65]. There are more than a hundred sign languages in the world, most of which have datasets in video form. Therefore, conducting advanced research to address the anticipated surge in data volume in the future is of paramount importance. In this work, we aim to enable the model to generate sign languages across diverse linguistic backgrounds, ensuring its adaptability to new datasets. More background information and related work can be found in supplementary materials.

[1] https://how2sign.github.io/

3 Our Benchmark: Prompt2Sign

The data collected and its format in previous work had some shortcomings, when there is a mismatch between these models in Sign Language Production (SLP) [10,71,73,83,96,97] and Sign Language Translation (SLT) [3,6,14,43], it can lead to complex challenges. For instance: (1) The results of the SLT model cannot be used as training data for the SLP model (*e.g.*, [5,87] & [70,75], recording methods is too different). (2) The results of the SLP model are hard to be used as input for the SLT model (evaluation experiment needs, *e.g.*, [5] & [3]). (3) The output of the SLP model is not suitable as input for most style transfer models (*e.g.*, [10,98,107], the researchers have to train a pose2video model themselves).
Data Collection. Compared to previous data collection, our process involves: downloading sign language videos of a specific language from the internet or public datasets [20, 26, 32, 58, 68, 81]; Then editing and aligning those videos; using OpenPose [8] to extract each frame of them into 2D keypoint json files and 2D keypoint visualized videos; The json keypoint files serve as input for our standardized data processing pipeline. Currently, Prompt2Sign covers eight sign languages, we show a detailed comparison of datasets in Table 1.
Data Format. Building on previous work [71, 75, 104], we have developed a three-step tool for standardizing data processing. The tool is highly efficient, and lightweight, requiring no additional model loading. So it is extremely friendly for large-scale data processing. Furthermore, we improved it specifically for sign language (*e.g.*, the removal of unnecessary computations such as leg movements, and embedded it into the processing tool of Prompt2Sign. The key steps we defined include "json (2D Keypoints) to h5", "h5 to txt (3D Keypoints)", and "txt to skels (Standard Pose Storage, Data of Keypoints)". Among all the steps, the most crucial part is the transition from 2D to 3D Pose (*i.e.*, training needs more accurate 3D positioning of fingers, rather than overlapping 2D fingers):

`Step I`: First, we obtain the length of the skeleton through the 2D keypoint coordinates (x and y), a and b represent indices that identify the two keypoints (or joints) forming a bone $L = \sqrt{(ax - bx)^2 + (ay - by)^2}$.
`Step II`: We compute the 3D rotation angles by integrating the aforementioned data, A stands for angles: $A_x, A_y, A_z = \frac{\text{angle}_x, \text{angle}_y, \text{angle}_z}{\sqrt{\text{angle}_x^2 + \text{angle}_y^2 + \text{angle}_z^2}}$
`Step III`: In the 3D space (x, y, z), Y_x, Y_y and Y_z represent the current coordinates of the Joint $Y_x, Y_y, Y_z = Y_x + L \times A_x, Y_y + L \times A_y, Y_z + L \times A_z$.

These mathematical formulas initialize the skeletal model, including the calculation of skeletal length L, root node position, rotation angle, and 3D coordinates (x, y, z), as the subsequent input for the simulation of 3D human skeletal motion. Compared to previous work [104], it is more concise and elegant, eliminating 60% of the bloated processes. In this way, we extract the posture/gesture information that the sign language only cares about from the video file. *The purpose of the transformation is to discard redundant information, compress the size (80% reduction in size compared to the clip video in the ASL part), and unify the reading format so that more text2text and seq2seq models/applications can use the datasets, rather than develop data loaders dedicated to sign language.*

Dataset Statistics. After the data processing, train, dev, and test sets of different language parts are shown in Table 2. We constructed 120 English templates and 210 prompt word templates for other languages (with 30 templates for each language), which were randomly associated with the script data to form a part of our dataset. This prompt word data is needed for the future development of large language models of sign language because we were able to develop a model with understands more complex, natural human conversational inputs by using prompt word data. More data structures and prompt word examples can be found in supplementary materials.

Table 2: Dataset Statistics [20, 26, 32, 58, 68, 81]: The number of video clips in their train, dev, test set, respectively.

Subset	ASL	GSL	DSGS	LSF-CH	LIS-CH	LSA	KSL	TSL
Train	31,047	7,096	8,043	5,672	2,254	2,400	17,000	28,142
Dev	1,739	519	500	500	250	400	2,000	4,418
Test	2,343	642	500	250	250	400	6,000	3,742

4 Our Model: SignLLM

In this section, we first introduce our multilingual SLP work in this Sec. 4.1, the traditional SLP method in Sec. 4.2, the fundamental architecture of SignLLM in Sec. 4.3-4.4, Finally, we highlight the specific alterations made and the implications and benefits of modifying the traditional method.

4.1 Design Overview

Basic Idea. SignLLM has two modes named Multi-Language Switching Framework (MLSF) and Prompt2LangGloss, as shown in Figure 2 (middle), both make the model capable of multilingual sign language production by using the multilingual Prompt2Sign dataset. SignLLM also has a novel loss function, accompanied by a corresponding functional component for accelerated training.
Motivation. The motivations behind MLSF and Prompt2LangGloss differ, the MLSF aims to achieve multilingual SLP without causing semantic confusion, but Prompt2LangGloss strives to achieve user-friendliness and enhance the model's understanding of complex inputs while enabling multilingual SLP. However, both modes share a common motivation, which is to establish a paradigm that encourages other researchers to explore multilingual generation and recognition using our new dataset and promote the conversion of existing models into multilingual models. The introduction of RL concepts is aimed at accelerating the training in our larger dataset, and both multilingual SLP modes can benefit from it.

4.2 Preliminary of Text2Pose Method

The general SLP pipeline (*i.e.*, text to sign language video) [37, 71–74] has following steps: text-to-gloss conversion, gloss-to-pose mapping, and finally pose-to-video rendering. In our work, we mainly focus on/modify the first two steps:
Text2Gloss & Gloss2Pose. Essentially, the transformation from text-to-gloss and gloss-to-pose can be distilled to a sequence-to-sequence [71, 72] problem in the realm of textual data, and their structures to bear significant resemblances. We define the x_u ($\boxed{x_u}$) as the input text x tokens at position u (total number is $\mathcal{U}$, position from 1 to $\mathcal{U}$), the p_w ($\boxed{p_w}$) as the output pose p at position w (total

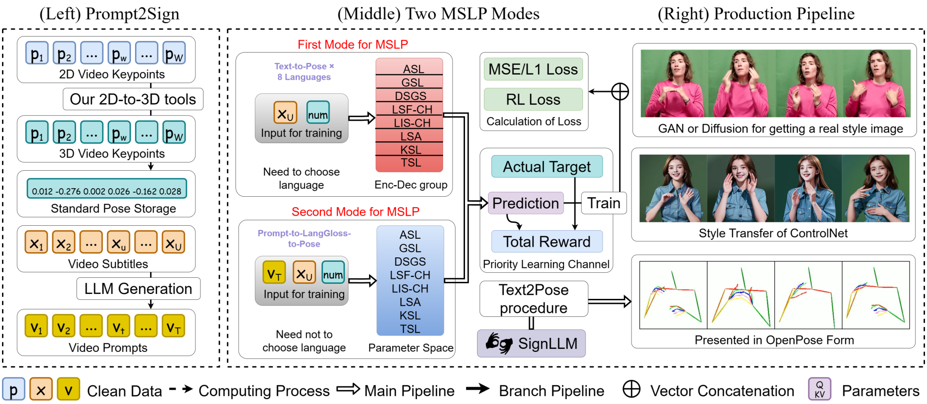

Fig. 2: (Left) Data types and abstract representations of PROMPT2SIGN dataset. (Middle) The training process of Prompt2LangGloss and MLSF, the computational principle of Reinforcement Learning Loss. (Right) The output of SIGNLLM can be converted into most pose representation formats, which can then be rendered into realistic human appearance by style transfer/specifically fine-tuned generative model.

number is $\mathcal{W}$, frame position from 1 to W), and then we use an encoder-decoder transformer framework to convert input into output:

$$f_u = E_{input2output}(x_u | x_{1:\mathcal{U}}) \tag{1}$$

$$p_{w+1} = D_{input2output}(p_w | p_{1:w-1}, f_{1:\mathcal{U}}) \tag{2}$$

Here, f_u denotes the encoded source of x_u. The output text tokens generated from this process form the input for the next stage of our translation model. In short, text2pose is the core method of SLP, and some people use gloss as an intermediate conversion medium to make it text2gloss2pose.

4.3 Two Multilingual SLP Modes

Implementation Overview. For the two modes of SIGNLLM, their difference is that the former is parallel to the Enc-Dec (*i.e.*, Text2Pose × number of languages), while the latter adds a marker at the gloss channel (*i.e.*, Text2Gloss2Pose → Prompt2LangGloss2Pose).

Multi-Language Switching Framework. The MLSF framework can be understood as having multiple parallel Text2Pose channels/Enc-Dec, each language has an Enc-Dec, allowing each channel/Enc-Dec to be independently trained and inferred. Text2Pose visual representation is shown in the middle of Figure 2, the red rectangle represents the eight Enc-Dec in our model, and the middle partition represents the parameters separated and not confused with each other.

The assignment operation could be formalized as $\text{Enc}_\mathcal{L} = \mathcal{E}_\mathcal{L}$ and $\text{Dec}_\mathcal{L} = \mathcal{D}_\mathcal{L}$. Here, $\mathcal{L}$ denotes the language of input, while $\mathcal{E}_\mathcal{L}$ and $\mathcal{D}_\mathcal{L}$ are the mapping from language $\mathcal{L}$ to an encoder and decoder in the sets $\mathcal{E}$ and $\mathcal{D}$. The framework incorporates wider sign languages into the model, enhancing the flexibility and scalability, it is often convenient for users and researchers, which not only fosters activity in the field but also accelerates research advancement.

Table 3: The difference between the two modes: M and P represent MLSF and Prompt2LangGloss, respectively. Adress represents which traditional step has been innovated, while Features represents the ability of the mode to focus more on.

Mode	Function	Address	Enc-Dec	Prompt	Feature	Note
M	multilingual SLP	text2pose	Multiple	No	More efficient/stable	Language is easy to add or subtract
P	multilingual SLP	text2gloss	Single	Allow	Understand complex input	Greater potential for development

Prompt2LangGloss. In contrast to MLSF's major framework modifications, Prompt2LangGloss represents the amendments made during the development stages of Text2Gloss and Gloss2Pose. It is the second multilingual training strategy, to make the model understand complex inputs such as prompt words. Additionally, it utilizes LangGloss instead of the manual annotation gloss (automatically annotates inside the model).

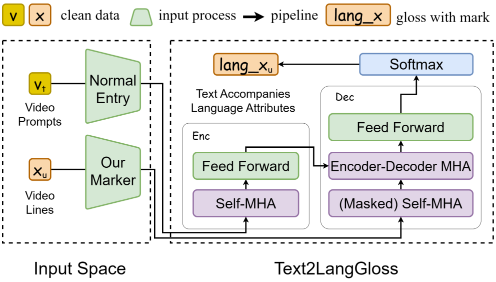

Fig. 3: We enhance Text2Gloss [71] with a marker to generate the Gloss with linguistic properties. The $\boldsymbol{v}_t$ (v_t) and $\boldsymbol{x}_u$ (x_u) represent data types and abstract representations.

Gloss, essentially a textual representation of sign language gestures, operates as an intermediate entity when using a text-to-pose (T2P) model. As shown in Figure 3, our proposed enhancement of this model involves appending an additional language attribute to each Gloss during the reading and tokenizing stages. For instance, a traditional gloss token "<xxx>" can be transformed into "<ASL_xxx>", thus introducing a layer of conditional input $f_u = E_{T2LG}(x_u|x_{1:U})$ into SLP based on Eq. 1: $lg_{w+1} = D_{T2LG}(lg_w|lg_{1:w-1}, f_{1:\mathcal{U}})$.

In this way, we solved several challenges: (1) LangGloss makes the existing models able to do multilingual data training. And by adding language attributes to gloss, we reduce the potential for semantic confusion caused by the same meanings of words in various languages. (2) LangGloss as a mediator can solve the limitations of the existing model in understanding complex, natural human inputs. It reduces the negative impact of directly processing intricate prompt words, thereby improving the model's accuracy. (3) By using LangGloss, we significantly reduce the reliance on costly and time-consuming manual annotations for gloss. It incorporates an automatic gloss marker within the model, reducing the need for extensive manual data compared to previous approaches.

Discussion on Two Modes. MLSF dynamically adds encoders, which can avoid semantic confusion and maximize its convenience (*e.g.*, a general model can execute multilingual SLP tasks that were impossible for researchers in the past. It saves significant development time, potentially twice the effort, ten times the return). Prompt2LangGloss is more focused on improving the ability to understand complex inputs, which is complementary to the MLSF. It will have great prospects with the data volume increase (*e.g.*, ChatGPT rarely mixes languages when speaking in a specific language). Moreover, LangGloss can be used without choosing a language as a valuable feature. Therefore, both approaches have their focus, input type and user-cases, summarized in Table 3.

4.4 Reinforcement Learning for Accelerated Training

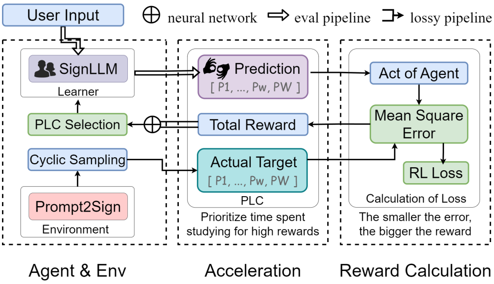

Fig. 4: We use some elements: `User`, `Agent`, `Environment`, `Iterative Update Process`, `PLC` to sketch the reinforcement learning process adapted for sequence prediction.

Reinforcement Learning Loss. Because **R**einforcement **L**earning has an important property in calculating which actions/samples are more valuable. So selective training data could solve the challenge of slow training using more sign languages and larger datasets. But before selecting, we should first transform the ordinary generative model into an RL-like model, so we design **R**einforcement **L**earning Loss for model transformation.

Concretely, we set the input sequence as the state s_t, the output sequence is the action a_t, and the reward r_t. t stands for time, i stands sample. The closer the prediction is to reality (mean squared error), the greater the reward: $r = -\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$. With this interpretation, we can reformulate the traditional supervised learning problem of minimizing MSE loss to maximize the expected cumulative reward, where L denotes the MSE loss function, M is the model, and x_t, y_t are the model inputs and corresponding targets respectively:

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, \mathbb{E}_\theta\left[\sum_{t=0}^{T} r_t\right] = \underset{\theta}{\operatorname{argmin}}\, \mathbb{E}_\theta\left[\sum_{t=0}^{T} L(y_t, M(x_t))\right] \tag{3}$$

Optimized parameters θ are found using gradient descent, updating parameters proportionally to the gradient of expected cumulative reward concerning model parameters. The whole process is shown in Figure 4, *which provides a robust mathematical framework for sequence learning tasks and allows for the integration of sophisticated optimization strategies from RL.*

Priority Learning Channel. The RL Loss itself does not possess subjective acceleration capabilities, as we said. It is designed for **P**riority **L**earning **C**hannel (PLC) to prioritize the learning of more valuable data. Last paragraph, we have defined rewards r, sample i, and data for each batch j, which then are converted into sampling probabilities for each data sample according to $P(i) = \frac{r(i)^\eta}{\sum_{j\in S} r(j)^\eta}$, where η regulates the intensity of prioritization, and S represents the dataset. By employing these sampling probabilities, the choice of data samples for each batch is no longer uniform but regulated by their respective rewards. The per-instance RL loss, $L(i)$, is computed for the chosen instances, which is then used to optimize the model parameters following the policy gradient theorem. This procedure is formally expressed as $Minimize \ \ E_{i\sim P(i)}[L(i)]$. By continually updating the model based on the most rewarding samples, the PLC brings the advantages of RL to supervised learning for sequence prediction tasks. *The adaptive nature of the PLC ensures that the model's focus shifts by the model's evolving knowledge, thereby accelerating the learning process.*

Table 4: American Sign Language Production (ASLP): Comparison of SIGNLLM variants with baseline on *Text to Pose* task by using our PROMPT2SIGN ASL part. The blue areas and the gray areas stand for the result of two modes.

Type:	DEV SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	TEST SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
ASL-SLP Transformers [23]	17.19	23.11	29.49	36.96	55.85	12.85	17.35	23.38	39.46	46.89
SignLLM-1x40M-Base-M (ASL)	18.77	25.42	32.44	40.66	61.44	14.13	19.08	25.72	43.40	51.57
SignLLM-1x120M-Large-M (ASL)	19.40	26.11	33.38	41.89	63.21	14.52	19.65	26.44	44.72	53.14
SignLLM-1x1B-Super-M (ASL)	**20.09** +2.9	**27.04**	**34.45**	**43.16**	**65.19**	**15.03** +2.18	**20.28**	**27.35**	**46.17**	**54.86**
SignLLM-1x40M-Base-P (ASL)	17.34	23.57	29.87	37.81	56.93	13.06	17.66	23.77	40.15	47.76
SignLLM-1x120M-Large-P (ASL)	18.05	24.28	31.04	38.97	58.78	13.48	18.27	24.57	41.57	49.42
SignLLM-1x1B-Super-P (ASL)	**18.68** +1.49	25.11	31.99	40.14	60.47	**13.93** +0.92	18.86	25.40	42.87	50.91

5 Experiments, Evaluations, and Discussions

Setup. The encoder and decoder components of our all encoder-decoder model versions (*i.e.*, Base, Large, Super) both have two layers. Each layer has four attention heads to improve modeling capabilities. The Base version model parameter settings include an embedding dimension of 512, a hidden size of 512, and a feed-forward size of 2048. When the model is expanded to Large and Super versions, the number of layers and attention heads remain the same, while the other parameters are expanded by about two and four times, respectively. This architecture allows for enhanced representation learning and attention mechanisms, enabling the model to handle more complex and demanding tasks. We evaluate our MLSF and Prompt2LangGloss modes. The former embodies a dynamically extended encoder-decoder architecture where parameters are not shared across languages. The latter uses a Prompt2LangGloss strategy for parameter sharing, gloss marking and incorporates prompt word fine-tuning.

For the model evaluation, we provide the naming rules as follows: SignLLM-{number of languages}x{single language parameters}-{submode size}-{the mode of training}-{the language of input}. Such as "SIGNLLM-2x40M-Base-M (ASL)", the nomenclature "2x40" denotes that the model comprises 2 language knowledge, with each language component estimated to be around 40 million paragramter in size, and a total is 80 million parameters ("1B" represents a total of 1 billion parameters). There are Base, Large, and Super versions, depending on a single language parameters size provided by the model. M and P stand for models trained using corresponding modes. At the end is the language of the input model, which is the sign language of what language to make the model output, ASL, GSL, *etc*. More details can be found in the supplementary materials.

Metrics. In this paper, we mainly use 3 evaluation metrics: `(i) BLEU-n score` measures the similarity between machine-generated translations and reference translations based on n-grams, the closer the predicted result is to the input (reference), the higher the value. BLEU-n [60] means that n words are used as the basic computing unit, and the higher the n, the higher the fluency requirement. `(ii) ROUGE score` [49] is similar to BLEU, but is more concerned with consistency and coverage. It indicates better agreement between the generated and reference texts, indicating a more accurate and comprehensive summary. `(iii) DTW score` underpinned by dynamic programming principles [2], is employed to ascertain the smallest manipulation distance between clips and sentences, the lower the better. Other metrics are mentioned in the Caption of the image.

Table 5: Multilingual Sign Language Production (MSLP): Comparison of different SignLLM M-mode variants with a baseline on *Text to Pose* task. We propose the first Multilingual SLP benchmark, with the exception of the existing TSL-Baseline

Type:	Language:	DEV SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	TEST SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
SignLLM-6x40M-Base-M	DSGS	9.73	15.82	19.85	24.84	37.57	7.34	9.89	16.85	26.04	31.51
SignLLM-6x120M-Large-M	DSGS	**11.45**	17.28	19.84	29.07	41.69	**9.69**	14.06	16.35	29.80	31.51
SignLLM-6x40M-Base-M	LSF-CH	9.79	18.48	23.13	28.86	34.98	8.92	13.12	16.11	26.48	37.92
SignLLM-6x120M-Large-M	LSF-CH	**13.72**	20.79	23.40	25.15	38.39	**9.60**	12.58	16.98	22.71	41.96
SignLLM-6x40M-Base-M	LIS-CH	10.81	14.46	19.93	24.55	35.83	7.34	10.56	15.24	22.73	36.42
SignLLM-6x120M-Large-M	LIS-CH	**12.10**	18.04	23.01	25.95	36.98	**9.30**	11.20	15.68	23.38	38.37
SignLLM-6x40M-Base-M	LSA	10.72	15.55	21.76	25.91	38.78	7.33	14.86	16.68	22.55	34.42
SignLLM-6x120M-Large-M	LSA	**11.69**	14.79	26.25	28.08	39.01	**8.21**	11.04	17.05	26.68	37.46
SignLLM-6x40M-Base-M	KSL	12.79	19.98	23.62	35.98	43.56	11.31	16.12	24.16	32.97	46.02
SignLLM-6x120M-Large-M	KSL	**17.55**	26.44	20.67	32.80	51.62	**13.75**	17.79	25.06	35.07	45.96
Hybrid Translation System [42]	TSL	-	-	-	-	-	12.64	18.28	31.48	53.17	-
SignLLM-6x40M-Base-M	TSL	14.53	19.86	29.93	36.86	58.01	13.23	17.80	25.39	39.30	57.03
SignLLM-6x120M-Large-M	TSL	**15.17**	21.70	31.73	38.86	71.10	**14.36**	18.74	26.96	43.21	57.12

5.1 Quantitative Evaluation

Back Translation. Back translation means that translate generated sign language videos back into spoken language sentences. These sentences are then compared with the original input sentences to evaluate the translation quality. This task is widely adopted to evaluate SLP as it can offer a good indication of the expressiveness of the produced sign language videos [71], translation models are trained on the corresponding language data.

Table 6: Performance of German SLP: Comparison of different models with SOTA work on *Text to Pose* task. The **increase number** is compared to the latest sign language production work.

Approach:	DEV SET BLEU-4	ROUGE	TEST SET BLEU-4	ROUGE
Progressive Transformers [71]	11.82	33.18	10.51	32.46
Adversarial Training [70]	12.65	33.68	10.81	32.74
Mixture Density Networks [72]	11.54	33.40	11.68	33.19
Mixture of Motion Primitives [73]	14.03	37.76	13.30	36.77
Photo-realistic SLP [75]	16.92	35.74	21.10	42.57
SignDiff SLP Transformers [23]	18.26	39.62	22.15	46.82
SignLLM-1x40M-Base-M (GSL)	18.61	40.69	22.76	48.05
SignLLM-1x120M-Large-M (GSL)	**19.31** +1.05	41.42	**23.25** +1.10	49.08
SignLLM-1x1B-Super-M (GSL)	19.07	**41.83**	23.21	**49.52**
SignLLM-1x40M-Base-P (GSL)	17.12	37.43	20.93	44.21
SignLLM-1x120M-Large-P (GSL)	17.55	38.10	21.39	45.16
SignLLM-1x1B-Super-P (GSL)	17.54	38.48	21.35	45.57

In Table 4, we conduct American Sign Language Production back-translation tests using SignLLM on the ASL part of our new dataset, and Table 6 further compares our method with other SOTA approaches for GSLP on the GSL part of Prompt2Sign dataset [23, 70–73]. These two languages stand for high-resource languages (*i.e.*, languages with rich data resources), and we have conducted tests at different levels on the dataset, all of which have shown impressive performance, compared with SOTA works in the field. This affirms the competitiveness and potential of our proposed method, regardless of the specific sign language in use.

In Table 5, we present the results of our model for six different sign languages. These six languages represent low-resource languages, and they are considered limited languages. For languages with low resources, their vocabulary, video time, and diverse corpus sources are relatively low, making training more difficult. From the table data, it can be observed that our performance remains strong in languages where training data is lacking. As long as the input text/prompt can be encoded as a computationally recognizable word and video exists, our method is capable of translating it into the corresponding language after training.

Table 7: Comparison of Different Modules: SignLLM-40M-Base results for *Text to Pose* task on the ASL part of Prompt2Sign, with multiple data augmentation techniques. Base: Multi-Language Switching Framework with Normal MSE Loss, no enhancement Settings. FP: Future Prediction. GN: Gaussian Noise. Prompt2LangGloss: Prompt2LangGloss with Normal MSE Loss. PLC: Priority Learning Channel. The **increase number** is compared to the base model. The blue areas are our results, the gray areas are for reference, and the other lines are traditional techniques. Top performances are highlighted in **bold**, while second top performances are underlined.

	DEV SET					TEST SET				
Approach:	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Base	10.96	14.68	22.49	40.29	44.13	9.27	10.72	18.64	39.88	40.39
Future Prediction	13.48	18.64	27.69	46.88	48.45	13.76	16.21	25.88	44.57	46.47
Just Counter	15.29	21.98	29.72	48.54	50.95	13.73	17.02	27.65	47.71	48.79
Gaussian Noise	16.28	23.53	32.05	50.76	53.68	15.63	20.35	29.19	47.91	50.16
FP & GN	16.44	23.79	32.32	50.97	53.44	**17.17**	**21.25**	**30.12**	**50.25**	50.47
Prompt2LangGloss	15.78	22.66	30.27	48.30	50.88	16.32 **+7.05**	20.27	28.70	47.89	48.18
RL Loss	18.33	24.28	32.18	48.83	52.61	13.26	17.33	24.51	40.76	40.95
RL Loss & PLC	**18.77 +7.81**	**25.42**	**32.44**	**52.66**	**61.44**	14.13	19.08	25.72	43.40	**51.57**

5.2 Ablation Evaluation

Performance. Table 7 indicates our last two innovative strategies significantly improve the model's performance, our new RL Loss and PLC technique, contributing to substantial improvements. In addition, despite the demonstrable improvement RL Loss facilitates in the development set, its effectiveness is less apparent in the test set. However, a completely different picture emerges when we combine RL Loss with the PLC. This innovative integration led to a considerable improvement in the ROUGE score on the test set, showcasing the positive impact of our proposed technique. We speculate that this is due to *"Reinforcement learning is weak in unknown environments, whereas PLC enhances adaptability."*

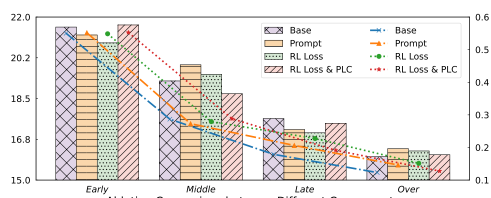

Fig. 5: Training Efficiency: Comparison of different Settings on DTW values (the lower the better) at different epoch times. Left Y-axis: Value of DTW. Right Y-axis: Value of Loss. Prompt: Prompt2LangGloss mode.

In Figure 5, we compare the performance of the Base model with different module schemes. The comparison is primarily based on observing the variability of DTW scores across epochs to assess the effectiveness of each approach. We observe that: (i) The standalone use of the Prompt2LangGloss model exhibits the lowest efficiency, as it introduces noise by incorporating prompts and tokenizers, which is also evident in Table 7. (ii) The combination of the two RL methods shows modest performance improvement compared to using each method individually although it is not statistically significant, suggesting that there is still potential for enhancing this approach further. Exploring optimization techniques for each RL method could be considered to achieve better results. (iii) In the most noteworthy aspect of reducing training time, they have played a significant role. Compared to the Base setting, using the RL Loss with PLC setting achieves the "Late" stage in just 72.9% of the time.

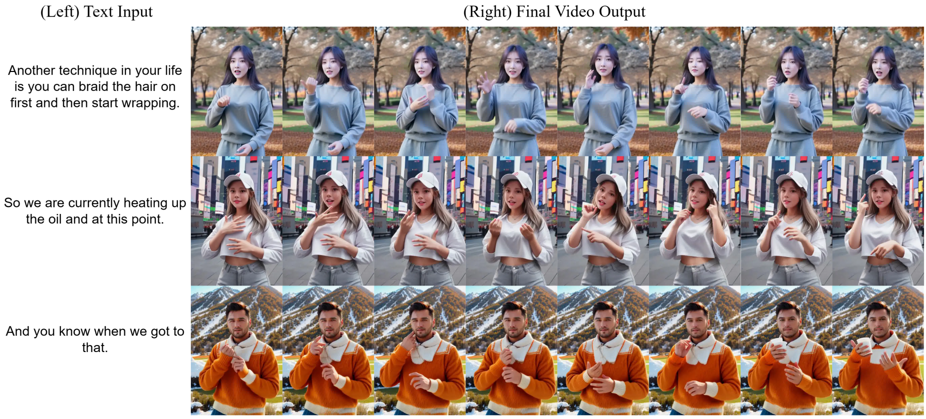

Fig. 6: (Left) Texts or Prompts as model inputs. (Right) We use an adjusted style transfer model [24] to convert the predicated pose video into the final video.

Table 8: Prompt Channel Accuracy: We investigate the information loss of Prompt input into the Prompt2LangGloss channel by German SLP.

	DEV SET		TEST SET	
Approach:	BLEU-4 ↑	ROUGE ↑	BLEU-4 ↑	ROUGE ↑
Stoll *et al.* [82]	16.34	48.42	15.26	48.10
Baseline [71]	20.23	55.41	19.10	54.55
Ours	**23.10**	**58.76**	**22.05**	**56.46**
Δ Acc.	*+ 14.2%*	*+ 6.0%*	*+ 15.4%*	*+ 3.5%*

Prompt Fine-Tuning and User Study. By employing prompt words as input for the text channel and using the original text or the original gloss as input for the gloss channel (*i.e.*, usage of Prompt2LangGloss), we are able to develop a model with understanding prompts competency. This approach aims to translate natural language into objective text/gloss before inputting them into the model. In reality, users might question, *"How do you demonstrate 'the sky is blue' in sign language?"*, rather than directly inputting *"the sky is blue"*. This training strategy gives the model a degree of Understanding more complex input text ability. We compared the Prompt2LangGloss channel with a tokenizer to the previous Text2Gloss approach in terms of prompt usage. The experiments indicated that the impact of the tokenizer in the Prompt2LangGloss channel is relatively small and can be overcome through better training. As presented in Table 8, it underscores the effectiveness of our mode in reducing semantic information loss in the channel.

5.3 Qualitative Evaluation

Qualitative Presentation. We use our predicted pose results as input, which are then used to generate rendered sign language videos in Figure 6. We can observe that our video outcomes are of high quality, with highly accurate finger movements and high image fidelity. Our results surpass all previous works, benefiting not only from technological advancements but also from the superior output quality of our SignLLM compared to previous smaller models. The postures we predict are rarely missing, unlike previous works that often suffered from issues such as flickering, incomplete or missing fingers, and low input quality due to densely packed fingers. So it works well with the latest model [24].

In Table 9, we conducted a series of ablation tests and back-translation evaluations based on the German SLP task to investigate two main research questions: (i) Whether rendering the predicted results into real sign language videos would lead to a decrease in accuracy. (ii) How our work compares to previous studies in the task of text-to-sign-language real video generation. Based on our observations, there was generally not a significant accuracy loss, but there were some fluctuations compared to the baseline. Our approach outperformed previous works, which could be attributed to the higher quality of our data, making it more suitable as input for style transfer models.

Table 9: Presentation Effect Study: Comparison of different models with previous work on *Text to Sign* task in GSL.

Approach:	DEV SET BLEU-4	DEV SET ROUGE	TEST SET BLEU-4	TEST SET ROUGE
Progressive Transformers [71]	10.79	36.15	9.59	35.42
Baseline [23]	16.68	43.2	24.24	51.05
SignLLM-1x40M-Base-M (GSL)	16.96	44.41	24.74	52.45
SignLLM-1x120M-Large-M (GSL)	**17.73** +1.05	45.11	**25.39** +1.15	53.45
SignLLM-1x40M-Base-P (GSL)	16.27	43.76	24.12	54.35
SignLLM-1x120M-Large-P (GSL)	**16.71**	45.40	**25.04**	52.80

5.4 Discussion

Societal Impact. Our model has the potential to assist the people with disabilities in various areas, such as sign language teaching, generative sign language translation, and real-time interpretation for broadcasting. Firstly, current sign language teaching relies on human instructors and pictorial representations in books, which is not as convenient as learning textual languages [19]. Secondly, the existing sign language generative translation software is not good enough, making it difficult for family members of the deaf people to communicate without much knowledge of sign language. This requires them to consult pictorial dictionaries. Moreover, only a few videos, such as major news broadcasts, are accompanied by real-time sign language interpretation. It making it extremely challenging for the underserved community to lead a normal life. However, its current level of accuracy is not high enough to be fully trusted, and users must be cautious as improper usage may result in potential negative consequences.

Limitation. Although our tool has significantly advanced the automation of sign language data processing and data acquisition, it is not a complete end-to-end solution. *e.g.*, if someone wants to use his own private dataset, using OpenPose [8] to extract 2D keypoint json files and manual editing is necessary.

6 Conclusion

We present the first multilingual SLP model, SignLLM, based on a standardized multilingual sign language dataset, Prompt2Sign, that we have proposed. Our model with two modes, MLSF and Prompt2LangGloss, progressively incorporates a diverse more sign language and mitigates the issues caused by shared parameters. Our new loss and new module solve the issues of long training time due to larger datasets and more languages. Finally, we show baseline comparisons, ablation studies, experiments under various parameters, and qualitative evaluations for discussion, which proves the efficacy of our methodology.

References

1. Balakrishnan, G., Zhao, A., Dalca, A.V., Durand, F., Guttag, J.: Synthesizing Images of Humans in Unseen Poses. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
2. Berndt, D.J., Clifford, J.: Using Dynamic Time Warping to Find Patterns in Time Series. In: AAA1-94 Workshop on Knowledge Discovery in Databases (1994)
3. Boháček, M., Hrúz, M.: Sign pose-based transformer for word-level sign language recognition. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops. pp. 182–191 (January 2022)
4. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual (2020)
5. Cai, Z., Yin, W., Zeng, A., Wei, C., Sun, Q., Wang, Y., Pang, H.E., Mei, H., Zhang, M., Zhang, L., Loy, C.C., Yang, L., Liu, Z.: Smpler-x: Scaling up expressive human pose and shape estimation (2023)
6. Camgöz, N.C., Hadfield, S., Koller, O., Ney, H., Bowden, R.: Neural Sign Language Translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
7. Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., Sheikh, Y.A.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
8. Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., Sheikh, Y.: OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
9. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
10. Chan, C., Ginosar, S., Zhou, T., Efros, A.A.: Everybody Dance Now. In: Proceedings of the IEEE International Conference on Computer Vision (CVPR) (2019)
11. Chen, X., Jiang, B., Liu, W., Huang, Z., Fu, B., Chen, T., Yu, G.: Executing your commands via motion diffusion in latent space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18000–18010 (2023)
12. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., Garcia, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A.M., Pillai, T.S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Diaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K., Eck, D., Dean, J., Petrov, S., Fiedel, N.: Palm: Scaling language modeling with pathways. CoRR **abs/2204.02311** (2022)

13. Cihan Camgoz, N., Hadfield, S., Koller, O., Ney, H., Bowden, R.: Neural sign language translation. In: CVPR. pp. 7784–7793 (2018)
14. Cihan Camgöz, N., Koller, O., Hadfield, S., Bowden, R.: Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
15. Cooper, H., Bowden, R.: Large Lexicon Detection of Sign Language. In: International Workshop on Human-Computer Interaction (2007)
16. Cox, S., Lincoln, M., Tryggvason, J., Nakisa, M., Wells, M., Tutt, M., Abbott, S.: TESSA, a System to Aid Communication with Deaf People. In: Proceedings of the ACM International Conference on Assistive Technologies (2002)
17. Cui, R., Liu, H., Zhang, C.: Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
18. Deng, Y., Yang, J., Chen, D., Wen, F., Tong, X.: Disentangled and Controllable Face Image Generation via 3D Imitative-Contrastive Learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
19. Dickinson, J.: Sign Language Interpreting in the Workplace. Gallaudet University Press (2017)
20. Duarte, A., Palaskar, S., Ventura, L., Ghadiyaram, D., DeHaan, K., Metze, F., Torres, J., Giro-i Nieto, X.: How2Sign: A Large-scale Multimodal Dataset for Continuous American Sign Language. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
21. Duarte, A., Palaskar, S., Ventura, L., Ghadiyaram, D., DeHaan, K., Metze, F., Torres, J., Giro-i Nieto, X.: How2Sign: A Large-Scale Multimodal Dataset for Continuous American Sign Language. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
22. Ebling, S.: Automatic Translation from German to Synthesized Swiss German Sign Language. Ph.D. thesis, University of Zurich (2016)
23. Fang, S., Sui, C., Zhang, X., Tian, Y.: Signdiff: Learning diffusion models for american sign language production (2023)
24. Feng, M., Liu, J., Yu, K., Yao, Y., Hui, Z., Guo, X., Lin, X., Xue, H., Shi, C., Li, X., Li, A., Kang, X., Lei, B., Cui, M., Ren, P., Xie, X.: Dreamoving: A human video generation framework based on diffusion models (2023)
25. Forster, J., Schmidt, C., Hoyoux, T., Koller, O., Zelle, U., Piater, J.H., Ney, H.: RWTH-PHOENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC) (2012)
26. Forster, J., Schmidt, C., Koller, O., Bellgardt, M., Ney, H.: Extensions of the sign language recognition and translation corpus rwth-phoenix-weather. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). pp. 1911–1916 (2014)
27. Gong, J., Foo, L.G., He, Y., Rahmani, H., Liu, J.: Llms are good sign language translators (2024)
28. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In: Proceedings of the Advances in Neural Information Processing Systems (NIPS) (2014)
29. Grobel, K., Assan, M.: Isolated Sign Language Recognition using Hidden Markov Models. In: IEEE International Conference on Systems, Man, and Cybernetics (1997)

30. Gueuwou, S., Siake, S., Leong, C., Müller, M.: Jwsign: A highly multilingual corpus of bible translations for more diversity in sign language processing. arXiv preprint arXiv:2311.10174 (2023)
31. Güler, R.A., Neverova, N., Kokkinos, I.: Densepose: Dense human pose estimation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7297–7306 (2018)
32. Ham, S., Park, K., Jang, Y., Oh, Y., Yun, S., Yoon, S., Kim, C.J., Park, H.M., Kweon, I.S.: Ksl-guide: A large-scale korean sign language dataset including interrogative sentences for guiding the deaf and hard-of-hearing. In: IEEE International Conference on Automatic Face and Gesture Recognition (2021)
33. Hanke, T., Schulder, M., Konrad, R., Jahn, E.: Extending the public dgs corpus in size and depth. In: LREC2020 - Workshop on the Representation and Processing of Sign Languages. pp. 75–82 (2020)
34. Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., de Las Casas, D., Hendricks, L.A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., van den Driessche, G., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., Rae, J.W., Vinyals, O., Sifre, L.: Training compute-optimal large language models **abs/2203.15556** (2022)
35. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)
36. Huang, J., Zhou, W., Zhang, Q., Li, H., Li, W.: Video-based sign language recognition without temporal segmentation. In: AAAI (2018)
37. Huang, W., Pan, W., Zhao, Z., Tian, Q.: Towards Fast and High-Quality Sign Language Production. In: Proceedings of the 29th ACM International Conference on Multimedia (2021)
38. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-Image Translation with Conditional Adversarial Networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
39. Kadir, T., Bowden, R., Ong, E.J., Zisserman, A.: Minimal Training, Large Lexicon, Unconstrained Sign Language Recognition. In: Proceedings of the British Machine Vision Conference (BMVC) (2004)
40. Kaplan, J., McCandlish, S., Henighan, T., Brown, T.B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., Amodei, D.: Scaling laws for neural language models. CoRR **abs/2001.08361** (2020)
41. Karpouzis, K., Caridakis, G., Fotinea, S.E., Efthimiou, E.: Educational Resources and Implementation of a Greek Sign Language Synthesis Architecture. Computers & Education (CAEO) (2007)
42. Kayahan, D., Güngör, T.: A hybrid translation system from turkish spoken language to turkish sign language. In: 2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA). pp. 1–6 (2019). https://doi.org/10.1109/INISTA.2019.8778347
43. Ko, S.K., Kim, C.J., Jung, H., Cho, C.: Neural Sign Language Translation based on Human Keypoint Estimation. Applied Sciences (2019)
44. Koller, O.: Quantitative Survey of the State of the Art in Sign Language Recognition. arXiv preprint arXiv:2008.09918 (2020)
45. Koller, O., Forster, J., Ney, H.: Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. Computer Vision and Image Understanding **141**, 108–125 (Dec 2015)

46. Koller, O., Forster, J., Ney, H.: Continuous Sign Language Recognition: Towards Large Vocabulary Statistical Recognition Systems Handling Multiple Signers. Computer Vision and Image Understanding (CVIU) (2015)
47. Kowalski, M., Garbin, S.J., Estellers, V., Baltrušaitis, T., Johnson, M., Shotton, J.: CONFIG: Controllable Neural Face Image Generation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
48. Kreutzer, J., Bastings, J., Riezler, S.: Joey NMT: A Minimalist NMT Toolkit for Novices. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) (2019)
49. Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out. pp. 74–81. Association for Computational Linguistics (Jul 2004)
50. Liu, Y., De Nadai, M., Zen, G., Sebe, N., Lepri, B.: Gesture-to-Gesture Translation in the Wild via Category-Independent Conditional Maps. In: Proceedings of the 27th ACM International Conference on Multimedia (2019)
51. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. In: ACM TOG (2015)
52. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.L., Yong, M.G., Lee, J., et al.: Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172 (2019)
53. Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., Van Gool, L.: Pose Guided Person Image Generation. In: Advances in Neural Information Processing Systems (NIPS) (2017)
54. Mallya, A., Wang, T.C., Sapra, K., Liu, M.Y.: World-Consistent Video-to-Video Synthesis. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
55. Mazumder, S., Mukhopadhyay, R., Namboodiri, V.P., Jawahar, C.V.: Translating sign language videos to talking faces. In: Proceedings of the Twelfth Indian Conference on Computer Vision, Graphics and Image Processing. ICVGIP '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3490035.3490286, https://doi.org/10.1145/3490035.3490286
56. McDonald, J., Wolfe, R., Schnepp, J., Hochgesang, J., Jamrozik, D.G., Stumbo, M., Berke, L., Bialek, M., Thomas, F.: Automated Technique for Real-Time Production of Lifelike Animations of American Sign Language. Universal Access in the Information Society (UAIS) (2016)
57. Men, Y., Mao, Y., Jiang, Y., Ma, W.Y., Lian, Z.: Controllable Person Image Synthesis with Attribute-Decomposed GAN. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
58. Müller, M., Alikhani, M., Avramidis, E., Bowden, R., Braffort, A., Cihan Camgöz, N., Ebling, S., España-Bonet, C., Göhring, A., Grundkiewicz, R., Inan, M., Jiang, Z., Koller, O., Moryossef, A., Rios, A., Shterionov, D., Sidler-Miserez, S., Tissi, K., Van Landuyt, D.: Findings of the second WMT shared task on sign language translation (WMT-SLT23). In: Koehn, P., Haddow, B., Kocmi, T., Monz, C. (eds.) Proceedings of the Eighth Conference on Machine Translation. pp. 68–94. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.wmt-1.4, https://aclanthology.org/2023.wmt-1.4
59. Neidle, C., Vogler, C.: A new web interface to facilitate access to corpora: Development of the asllrp data access interface (dai). In: Proc. 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon, LREC (2012)

60. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA (Jul 2002). https://doi.org/10.3115/1073083.1073135, https://aclanthology.org/P02-1040
61. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic Differentiation in PyTorch. In: NIPS Autodiff Workshop (2017)
62. Prajwal, K.R., Mukhopadhyay, R., Namboodiri, V.P., Jawahar, C.: A lip sync expert is all you need for speech to lip generation in the wild. In: Proceedings of the 28th ACM International Conference on Multimedia. p. 484–492. MM '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3394171.3413532, https://doi.org/10.1145/3394171.3413532
63. Qiu, S., Anwar, S., Barnes, N.: Dense-resolution network for point cloud classification and segmentation. In: WACV. pp. 3813–3822 (2021)
64. Radford, A., Metz, L., Chintala, S.: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv preprint arXiv:1511.06434 (2015)
65. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog p. 9 (2019)
66. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research **21**(1), 5485–5551 (2020)
67. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10684–10695 (June 2022)
68. Ronchetti, F., Quiroga, F., Estrebou, C., Lanzarini, L., Rosete, A.: Lsa64: A dataset of argentinian sign language. XX II Congreso Argentino de Ciencias de la Computación (CACIC) (2016)
69. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., Ho, J., Fleet, D.J., Norouzi, M.: Photorealistic text-to-image diffusion models with deep language understanding. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems. vol. 35, pp. 36479–36494. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/ec795aeadae0b7d230fa35cbaf04c041-Paper-Conference.pdf
70. Saunders, B., Camgöz, N.C., Bowden, R.: Adversarial Training for Multi-Channel Sign Language Production. In: Proceedings of the British Machine Vision Conference (BMVC) (2020)
71. Saunders, B., Camgöz, N.C., Bowden, R.: Progressive Transformers for End-to-End Sign Language Production. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
72. Saunders, B., Camgöz, N.C., Bowden, R.: Continuous 3D Multi-Channel Sign Language Production via Progressive Transformers and Mixture Density Networks. International Journal of Computer Vision (IJCV) (2021)
73. Saunders, B., Camgöz, N.C., Bowden, R.: Mixed SIGNals: Sign Language Production via a Mixture of Motion Primitives. In: Proceedings of the International Conference on Computer Vision (ICCV) (2021)

74. Saunders, B., Camgoz, N.C., Bowden, R.: Skeletal Graph Self-Attention: Embedding a Skeleton Inductive Bias into Sign Language Production. arXiv preprint arXiv:2112.05277 (2021)
75. Saunders, B., Camgoz, N.C., Bowden, R.: Signing at scale: Learning to co-articulate signs for large-scale photo-realistic sign language production. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5141–5151 (June 2022)
76. Schembri, A., Fenlon, J., Rentelis, R., Reynolds, S., Cormier, K.: Building the british sign language corpus. Language Documentation & Conservation **7**, 136–154 (2013)
77. Segouat, J.: A Study of Sign Language Coarticulation. ACM SIGACCESS Accessibility and Computing (2009)
78. Shanahan, M.: Talking about large language models. CoRR **abs/2212.03551** (2022)
79. Shi, B., Brentari, D., Shakhnarovich, G., Livescu, K.: Open-domain sign language translation learned from online video. Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (2022)
80. Siarohin, A., Sangineto, E., Lathuiliere, S., Sebe, N.: Deformable GANs for Pose-Based Human Image Generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
81. Sincan, O.M., Keles, H.Y.: Autsl: A large scale multi-modal turkish sign language dataset and baseline methods. IEEE Access **8**, 181340–181355 (2020). https://doi.org/10.1109/ACCESS.2020.3028072
82. Stoll, S., Camgöz, N.C., Hadfield, S., Bowden, R.: Sign Language Production using Neural Machine Translation and Generative Adversarial Networks. In: Proceedings of the British Machine Vision Conference (BMVC) (2018)
83. Stoll, S., Camgöz, N.C., Hadfield, S., Bowden, R.: Text2Sign: Towards Sign Language Production using Neural Machine Translation and Generative Adversarial Networks. International Journal of Computer Vision (IJCV) (2020)
84. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1–9 (2015). https://doi.org/10.1109/CVPR.2015.7298594
85. Tang, H., Bai, S., Zhang, L., Torr, P.H., Sebe, N.: XingGAN for Person Image Generation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
86. Tang, H., Wang, W., Xu, D., Yan, Y., Sebe, N.: GestureGAN for Hand Gesture-to-Gesture Translation in the wild. In: Proceedings of the 26th ACM International Conference on Multimedia (2018)
87. Tarrés, L., Gállego, G.I., Duarte, A., Torres, J., i Nieto, X.G.: Sign language translation from instructional videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (2023)
88. Taylor, R., Kardas, M., Cucurull, G., Scialom, T., Hartshorn, A., Saravia, E., Poulton, A., Kerkez, V., Stojnic, R.: Galactica: A large language model for science. CoRR **abs/2211.09085** (2022)
89. Thies, J., Elgharib, M., Tewari, A., Theobalt, C., Nießner, M.: Neural voice puppetry: Audio-driven facial reenactment. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 716–731. Springer International Publishing, Cham (2020)

90. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: Llama: Open and efficient foundation language models. CoRR (2023)
91. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: MoCoGAN: Decomposing Motion and Content for Video Generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
92. Ventura, L., Duarte, A., Giró-i Nieto, X.: Can Everybody Sign Now? Exploring Sign Language Video Generation from 2D Poses. In: ECCV Sign Language Recognition, Translation and Production Workshop (2020)
93. Von Agris, U., Kraiss, K.F.: Signum database: Video corpus for signer-independent continuous sign language recognition. In: Workshop on Representation and Processing of Sign Languages. pp. 243–246 (2010)
94. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating Videos with Scene Dynamics. In: Advances in Neural Information Processing Systems (NIPS) (2016)
95. Wang, T.C., Liu, M.Y., Tao, A., Liu, G., Kautz, J., Catanzaro, B.: Few-shot Video-to-Video Synthesis. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
96. Wang, T.C., Liu, M.Y., Zhu, J.Y., Liu, G., Tao, A., Kautz, J., Catanzaro, B.: Video-to-Video Synthesis. In: Advances in Neural Information Processing Systems (NIPS) (2018)
97. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
98. Wei, D., Xu, X., Shen, H., Huang, K.: GAC-GAN: A General Method for Appearance-Controllable Human Video Motion Transfer. IEEE Transactions on Multimedia (2020)
99. Wu, Z., Hoang, D., Lin, S.Y., Xie, Y., Chen, L., Lin, Y.Y., Wang, Z., Fan, W.: MM-Hand: 3D-Aware Multi-Modal Guided Hand Generative Network for 3D Hand Pose Synthesis. arXiv preprint arXiv:2010.01158 (2020)
100. Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., Raffel, C.: mt5: A massively multilingual pre-trained text-to-text transformer. arXiv preprint arXiv:2010.11934 (2020)
101. Yin, A., Zhao, Z., Jin, W., Zhang, M., Zeng, X., He, X.: Mlslt: Towards multilingual sign language translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5109–5119 (2022)
102. Yu, L., Yu, J., Li, M., Ling, Q.: Multimodal inputs driven talking face generation with spatial–temporal dependency. IEEE Transactions on Circuits and Systems for Video Technology **31**(1), 203–216 (2021). https://doi.org/10.1109/TCSVT.2020.2973374
103. Zakharov, E., Shysheya, A., Burkov, E., Lempitsky, V.: Few-Shot Adversarial Learning of Realistic Neural Talking Head Models. In: Proceedings of the IEEE International Conference on Computer Vision (CVPR) (2019)
104. Zelinka, J., Kanis, J.: Neural Sign Language Synthesis: Words Are Our Glosses. In: The IEEE Winter Conference on Applications of Computer Vision (WACV) (2020)
105. Zhang, L., Agrawala, M.: Adding conditional control to text-to-image diffusion models (2023)

106. Zhang, S., Yuan, J., Liao, M., Zhang, L.: Text2video: Text-driven talking-head video synthesis with personalized phoneme - pose dictionary. In: ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2659–2663 (2022). https://doi.org/10.1109/ICASSP43922.2022.9747380
107. Zhou, Y., Wang, Z., Fang, C., Bui, T., Berg, T.: Dance Dance Generation: Motion Transfer for Internet Videos. In: Proceedings of the IEEE International Conference on Computer Vision Workshops (2019)
108. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2017)
109. Zhu, Z., Huang, T., Shi, B., Yu, M., Wang, B., Bai, X.: Progressive Pose Attention Transfer for Person Image Generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
110. Zwitserlood, I., Verlinden, M., Ros, J., Van Der Schoot, S.: Synthetic Signing for the Deaf: Esign. In: Proceedings of the Conference and Workshop on Assistive Technologies for Vision and Hearing Impairment (CVHI) (2004)

SignLLM: Sign Languages Production Large Language Models
– Supplementary Material –

Sen Fang[1], Lei Wang[2,3], Ce Zheng[4]
Yapeng Tian[5], and Chen Chen[6]

[1]Rutgers University, [2]Australian National University, [3]Data61/CSIRO, [4]Carnegie Mellon University, [5]University of Texas at Dallas, [6]University of Central Florida
sen.fang@rutgers.edu

A Abstract of Materials

Below we provide more details, experimental results, and discussion. More details are in the https://signllm.github.io/.

B Background Information

Here we expand on some of the nouns mentioned on the main pages:
Gloss: In the context of sign language, gloss refers to the process of providing a word-for-word translation of sign language into written or spoken language. It involves assigning a specific written or spoken word to each sign in order to facilitate communication and understanding between sign-language users and non-sign-language users. It generally represents a specific gesture or posture.
OpenPose: OpenPose[1] is a real-time multi-person keypoint detection library that uses computer vision techniques to identify and track human body movements. The output result is a video of the key point visualization and key point data stored in json format for each frame (about 24 frames a second).
DensePose: DensePose[2] is a method that estimates dense correspondences between a 2D image and a 3D human model. It can be used to extract detailed information about the body posture, position, and movements of sign language users from 2D images or videos, stored or displayed as a dense map covering the entire body of a human being. Details can be found in the footnote links.

B.1 More Related Work

Here, we introduce the third step of sign language production: Pose2Video, which involves visualizing key points in a video rendering or converting it into a live person/model demonstration of sign language. We also give some basic concepts of reinforcement learning for a better understanding.

[1] https://github.com/CMU-Perceptual-Computing-Lab/openpose
[2] https://github.com/facebookresearch/detectron2/tree/DensePose

Rendering of Conditional Input. Conditioning refers to the capacity of a generative model to manipulate its output based on our intentions. Previous instances of conditional input Generative Adversarial Networks (GANs) [28] have exhibited favorable performance in generating images [38, 64, 97, 108] and videos [54, 91, 94–96]. Numerous studies have also focused on generating human poses while considering various factors, including entire body [1, 10, 53, 57, 80, 85, 109], face [18, 47, 89, 102, 103, 106], and hand [50, 86, 99]. One particular application is human-style transfer [69], which involves replacing a person in a video with another individual while preserving their actions. This technique has also found extensive use in sign language production [10, 98, 107]. The key aspect lies in extracting keypoints to replicate movements [10, 92], utilizing tools such as OpenPose, i3D, and DensePose for common keypoint extraction [10, 62, 98, 107]. In our work, we do not care about Pose2video, we only present some qualitative results at the end of the paper and in the supplementary materials.
Reinforcement Learning. It is the training or fine-tuning of large models is a common strategy. At the heart of reinforcement learning is the concept of a Markov Decision Process (MDP), an extension of Markov chains, which involves a finite set of states, a finite set of actions, state transition probabilities, and a reward function. The MDP delineates the interaction between an intelligent agent and the environment, wherein the agent chooses actions based on various states, and the environment imposes rewards or penalties on the agent based on the action and the current state, leading to a transition to the next state. An optimal policy is the mapping from state s to action a that maximizes the total expected return:

$$\pi^* = \arg\max_{\pi} \mathbb{E}[G_t | s_t = s, \pi] \tag{1}$$

where $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$, $0 \leq \gamma < 1$ is the discount factor, and $\mathbb{E}[]$ is the expectation operator. In LLMs, researchers often fine-tune models with reinforcement learning based on human feedback. Given that the SLP process aligns with the definition and can be reformed by the MDP, we simply simulate this concept to fine-tune our generation model. However, since the training scenario of sign language does not involve interaction with the environment, our reinforcement learning strategy is not a typical one, but rather only partially applied to component modules.

C More Details of Prompt2Sign

C.1 Modalities

In comparison to the previous datasets, we possess numerous additional advantageous attributes and a larger scale. As with the previous dataset work, we extracted everything automatically except speech/text. But we've added some automated channel tools that go deeper than that.
Prompt Word Templates. We constructed 120 English templates and 210 prompt word templates generated by GPT4 for other languages (with 30 tem-

Prompt Template & Some Examples
Part I
I really want to learn how to say "{Text}" in sign language. Can you help me?
How would you express "{Text}" in sign language?
Can you show me how to say "{Text}" in sign language?
How do I say "{Text}" in sign language?
Could you tell me how "{Text}" is represented in sign language?
What is the sign language for "{Text}"?
Part II
How is "So we're going to go up and down; let's switch hands, down and up; down and up." denoted in sign language?
Can you elucidate how "And just let those fingers relax." looks in sign language?
I really want to learn how "Now together you're going to go opposite." is said in sign language. Can you help?
How do I articulate "It's real easy to actually get your fingers to lead, so try not to let them do that." using sign language?
I am intrigued to learn the sign language for "Let the wrist do all the leading."
I am wondering how "Don't let the fingers take over, let the wrist do all the guiding." appears in sign language.
Part III
Ich möchte wirklich lernen wie man "{Text}" in Gebärdensprache sagt. Können Sie mir helfen?
Wie würden Sie "{Text}" in Gebärdensprache ausdrücken?
Können Sie mir zeigen wie man "{Text}" mit Gebärdensprache sagt?
Wie sage ich "{Text}" in Gebärdensprache?
Könnten Sie mir sagen wie "{Text}" in Gebärdensprache dargestellt wird?
Was ist die Gebärdensprache für "{Text}"?
Part IV
'Regen und Schnee lassen an den Alpen in der Nacht nach, im Norden und Nordosten fallen hier und da Schauer, sonst ist das klar' Wie stellt man das in Gebärdensprache dar?
Wie wird 'Sonnig geht es auch ins Wochenende, Samstag ein herrlicher Tag mit Temperaturen bis siebzehn Grad hier im Westen' in Gebärdensprache dargestellt?
Wie würden Sie 'Deutschland liegt morgen unter Hochdruckeinfluss, der die Wolken weitgehend vertreibt' gebärden?
Können Sie mir zeigen, wie 'am Sonntag im Nordwesten eine Mischung aus Sonne und Wolken mit einigen zum Teil gewittrigen Schauern' in Gebärdensprache aussieht?
Wie sieht die Gebärdensprache für 'örtlich Schauer oder Gewitter, die heftig sein können' aus?
Was ist die Gebärdensprache für 'und zum Wochenende wird es dann sogar wieder ein bisschen kälter'?

Table 1: We provide two templates for sign language as a reference, and `{Text}` is where the video oral dialogue is inserted.

plates for each language), which were randomly associated with the script data to form a part of our dataset. Some examples are in Table 1 below.
Data Enhancement. With tools that rewrite lines or prompt words, users can obtain several times more data to enhance the robustness of the trained model.
Multiview. Our multiple perspectives depend on the original video, and it is worth noting that if the researchers cannot guarantee that the newly acquired perspectives are all positive, then the model will generally be contaminated.
Depth Data. Our depth depends on whether the raw data video has relevant support, we believe that this is generally not needed, as most work uses lifting work to obtain 3D key points, rather than high-cost professional equipment.
Speech. Some of our audio comes from raw data and some comes from Google's text-to-audio tool. But that's not the main content of our dataset.
Compress. It refers to whether the data set has been compressed in a special way to make it easy to use. More like preprocessing, not compression like .zip.

C.2 Pose Information

Necessity of Uniform Standards. If there is a mismatch between any of these components in SLP [10,71,73,83,96,97] or SLT [3,6,14,43], it can lead to complex challenges. For instance, if the results of pose recognition cannot be used as training data, the results of SLR cannot be used for model testing, or if the results of sign language generation cannot be used as conditional input [10, 98, 107], which have troubled many novice researchers in the field.
Data Format Conversion.

– **How to extract key points?** We extracted two-dimensional (2D) frontal human pose information from videos of different resolutions, including upper body pose information of the body and hands, through OpenPose [7]. Includes 8 upper body key points. 21 keypoints in each hand, which is a total of 42 hand keypoints. These two parts add up to fifty keypoints, each of which has three XYZ messages, or 150 numbers.

Then in steps "json (2D keypoints) to h5", "h5 to txt (3D keypoints)", and "txt to skels (Standard Pose Storage)":

– **How to complete "json to h5"** We successively obtain a json number in a folder (a frame of pose information, 50 key points, 150 numbers), and then read all the json numbers in a folder into the key name of an h5 (h5 is a format of numpy) file, multiple folders form multiple build names, and finally form an h5 file.

– **How to complete "h5 to txt"?** We read each key name of h5 in turn (the original folder name), create the corresponding folder, each folder generates 5 txt files, the last one is the result, the first 4 txt stores the intermediate variable. This is the part of 2D to 3D, and the key formula of `Step III` in the text is the formula of this part. Additionally, we read the relevant

input : Three arrays Xx, Xy and Xw of size $T \times n$, a structure array, a sigma value for noise, a random number generator and a percentile value

output: Lines array, rootsx, rootsy, rootsz arrays, anglesx, anglesy, anglesz arrays, Yx, Yy, Yz arrays

```
Set T as number of rows and n as number of columns of Xx;
Initialize arrays lines, rootsx, rootsy, rootsz, anglesx, anglesy, anglesz;
Set rootsx as first column of Xx;Set rootsy as first column of Xy;
Set rootsz as array of zeros with size T;
Add noise to rootsx, rootsy, rootsz arrays;
Initialize arrays Yx, Yy, Yz as arrays of zeros with size T × n;
Set first column of Yx as rootsx; Set first column of Yy as rootsy; Set first
  column of Yz as rootsz;
for each bone in structure do
    Add empty list to lines;
    for each row t in range T do
        Compute length L using equation 1;
        Append L to lines;
    end
end
for each line in lines do
    Calculate max L as percentile of the line list;
    Assign math.log(max L) to lines array;
end
for each bone in structure do
    Assign a, b, line as elements of the bone;
    for each row t in range T do
        Compute rotation angles anglex, angley, anglez using equation 2;
        if any of anglex, angley, anglez is not finite then
            Set anglex, angley, anglez as 0.0;
        end
        if anglez < 0.0 then
            Set anglez as -anglez;
        end
        Add 0.001 to anglez;
        Normalize anglex, angley, anglez;
        Assign anglex, angley, anglez to anglesx[t, iBone], anglesy[t, iBone],
          anglesz[t, iBone];
        Compute new 3D coordinates Yx, Yy, Yz using equation 3;
    end
end
Reshape rootsx, rootsy, rootsz as arrays of size T × 1;
Return lines, rootsx, rootsy, rootsz, anglesx, anglesy, anglesz, Yx, Yy, Yz;
```

Algorithm 1: The core formula and code of 2D to 3D conversion.

data and delete the unqualified data such as NaN, 0, or replace it with the average median of the data. Finally, we condensed the data to about 1/5 of the original, this data comes from the processing of ASL part.

- **How to complete "txt to skels"?** We read the fifth txt file of each folder in turn, the number of lines in the txt file represents the number of frames of the folder corresponding to the video, we read a line of txt (150 numbers, separated by Spaces, a frame of information), plus a space, and then add a count value (the current line divided by the total number of lines, representing the progress bar), add a space after the count value, Then add the second line txt and continue to repeat the above. Then we put a txt (a piece of video information, the total number of numbers in it = 151* video frames) into a line of content, in this conversion process, tens of thousands of videos are all stored in our standard format.

C.3 More Details of the Data

Details of Processing. Firstly, we obtain the original video from the internet. As mentioned in the main text, this part still needs to be done manually, but a script can be written to speed up the process. What kind of sign language model you want to train requires corresponding corpus. Firstly, preliminary preprocessing can be done through scripts written by oneself or OpenASL [79] scripts . Secondly, the dialogue of the video is transcribed into text, videos are processed using OpenPose, and then used as input for our tool. Finally, enters the language mode corresponding to the data by setting the model to start training.
Time and Cost of Dataset Processing. Among all the data processing steps, the most time-consuming step is 2Dto3D, whose core code is shown in Algorithm 1, RTX3060 can process 1000 clips after 10 hours, and can process 50-80 hours of How2Sign data in about half a month (there is no 80 after editing). Improving the performance of a single card does not make it much faster, which may be caused by multithreading concurrency restrictions.

D More Experiments

D.1 Concrete Cases Study

We show specific production examples in both MLSF and Prompt2LangGloss Settings and in both ASL and GSL in Table 2 below. They include reference inputs and the results we produce and translate, as well as some intermediate results. In the case of ASL, we observe that the reference input contains the phrase "The number one loss for these birds is flight." Our model successfully generates the sign language translation, albeit with a slight variation: "The birds couldn't lose the flight." The generated output captures the meaning of the original text, showcasing the model's understanding of the semantic content. Similarly, in the case of GSL, the reference input includes the phrase "Hoher luftdruck sorgt bei uns morgen für viel sonnenschein" (High air pressure will bring a

Table 2: Concrete Cases Study: We select some sample examples for readers to understand better. When using the prompt, we referred to the intermediate text, which would have made the measurement more accurate. Although there is still a problem that the accuracy is reduced due to the loss in the process of prompt-word to text. Red and Blue mark similar part in the text in ASL and GSL, respectively.

Language	Format	Example	BLEU	ROUGE
(ASL)	Reference (Text) Prediction	The number one loss for these birds, is flight. The birds couldn't lose the flight.	20.21	62.86
(GSL)	Reference (Text) Prediction	Hoher luftdruck sorgt bei uns morgen für viel sonnenschein. Morgen viel sonnenschein wegen hohem luftdruck.	10.1	56.43
(ASL)	Prompt Reference (LangGloss) Prediction	What is the sign language for 'When does this take place?'. ASL_When ASL_does ASL_this ASL_take ASL_place? When does it takes up?	40.00	40.00
(GSL)	Prompt Reference (LangGloss) Prediction	Wie gebärdet man 'am freundlichsten wird es am meer' zeigen? GSL_Am GSL_freundlichsten GSL_wird GSL_es GSL_am GSL_meer. Am meer wird es am freundlichsten sein.	57.14	73.4

lot of sunshine for us tomorrow). Our model generates the translation "Morgen viel sonnenschein wegen hohem luftdruck" (Tomorrow, a lot of sunshine due to high air pressure), which accurately conveys the intended message. Multiple examples show that not only do we perform well with multilingual SLP, but there are exciting results with prompt word generation. The cases are adjusted for the sake of aesthetics, and the calculation result is based on the actual situation.

D.2 Extensibility & Visual Study

Subsequently, we provide an overview and comparison of motion capture techniques and novel visual models. Our objective is to advocate for the adoption of motion capture technology as a replacement for traditional visual methods in sign language rendering. Before that, we need to introduce some background:
SMPL skeleton system: The SMPL [51] (Skinned Multi-Person Linear) skeleton system is a parametric model that represents human body shape and pose. It is commonly used in computer graphics and animation. In the context of sign language, the SMPL skeleton system can be utilized to model and animate sign language movements and gestures.
VMD files and OpenMMD: VMD (Vocaloid Motion Data) files and OpenMMD (Open-source MikuMikuDance) refer to specific file formats and software tools used in character animation. VMD files contain motion data and are commonly used in the MikuMikuDance software for animating virtual characters. OpenMMD is an open-source implementation of the MikuMikuDance software that allows users to create and modify character animations. In the context of sign language, VMD files and OpenMMD can be utilized to animate virtual characters performing sign language gestures or movements.
Key point driven model: A key point driven model is a computational model or algorithm that relies on the detection and tracking of specific key points, landmarks, or features in order to analyze and interpret data or generate desired outputs. In the final pose-to-video stage of sign language rendering, the generation of realistic human videos from keypoints is essential. This can be accomplished through either motion capture or purely visual methods. In the following sections, we will evaluate the strengths and limitations of each approach. In the

	SSIM ↑	Hand SSIM ↑	Similarity ↑	F2FD ↓
Vid2Vid [96]	0.743	0.582	78.42	27.86
ControlNet [105]	0.817	0.646	82.11	25.47
Motion Capture	**0.826**	**0.687**	**81.29**	**22.71**

Table 3: Visual Study: SSIM: Comparison of image structure similarity between the generated image and the condition graph extracted from the Ground Truth. Similarity: Extract the similarity percentage of keypoints between the generated video and the input action. F2FD: The degree of difference between frames.

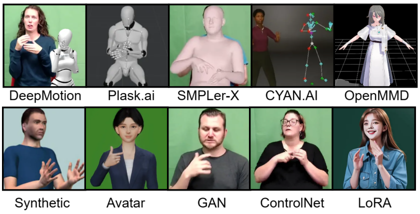

Fig. 1: Extensibility Presentation: We used five motion capture models and five sign language rendering models to show the final production effect.

context of sign language, a keypoints-driven model can be used to analyze and interpret sign language movements based on the detection and tracking of key points on the signer's body, such as hand positions, facial expressions, and body postures. It is our tentative exploration in this work.

D.3 Comparison of Motion and Visual

Extensibility Study. In Figure 1, the first line of it is obtained either directly or indirectly by reading our SignLLM output sequence through motion capture[3] [5, 11] software or models, while the second line of the image comes from the commonly used Pose2Vid [35, 54, 67, 91, 94, 105] or Pose2Img [38, 64, 97, 108] models. The broad scope of our model becomes apparent from the initial two statements. Subsequently, the next four lines present sign language demonstration videos created using either direct or indirect input of keypoints (some videos sourced from the project website). It is important to note that SMPLer-X and Avatar are utilized solely for demonstrative purposes in this context. Taking DeepMotion and VMD as instances, our model exhibits the capability to operate within a broader scope by utilizing keypoints as input, rather than relying solely on visual methods. This advancement provides the potential for more precise sign-language demonstrations. Details can be found in the footnote links.
Visual Study. We explored the influence of different forms on performance as shown in Table 3, current existing motion capture models do not fully support our keypoints format, and there may be some loss in certain transmission processes. Therefore, our primary focus is evaluating the presentation effect of motion capture models in sign language. Taking DeepMotion as an example, it is a deep learning-based method that drives models in a software environment using keypoints. In previous work, the comparison between rendered results and GroundTruth was measured using the structural similarity index (SSIM). However, since driving models do not have a specific GroundTruth, our comparison is based on the visualized keypoints extracted, which may introduce some errors but generally remain below 1%, providing a sufficient basis for simple comparisons. The percentage similarity refers to the comparison of extracted sequence numbers. Additionally, the difference between frames focuses on the smoothness

[3] DeepMotion; Plask.ai; Avatar; OpenMMD

Key name	Values	Note
BSLP method	{**<u>ASL</u>**, GSL}	Choose before training
Vocabulary size	{1k, 4k, 7k, **<u>16k</u>**}	Case-sensitive
Batch size	{**8**, <u>16</u>, 32}	Adjust according to configuration
Learning Rate (LR)[4]	{5e-2, **<u>1e-3</u>**, 5e-3}	Training initial value
Loss mode	{**<u>MSE</u>**, RL, L1, L2, LV}	Adjust according to situation
Max_sent_length	{<u>300</u>, **400**}	Input is usually less than maximum
Priority Learning Channel	{<u>False</u>, **True**}	Use with RL Loss
Dropout	{**<u>0</u>**, 0.1, 0.2, 0.3}	Adjust according to situation
# Layers (encoder-decoder)	{2-2, <u>4-4</u>, **8-8**}	Not necessarily correspond
Embed dim	{512, <u>1024</u>, **2048**}	Adjust for the amount of data
FFN dim	{2048, <u>4096</u>, **8192**}	Must equal to 4*hidden size
# Attention heads	{4, <u>8</u>, **16**}	Adjust for the amount of data

Table 4: Hyperparameters space: Optimal choices found during validation are marked in **bold**, while defaults are <u>underlined</u>. The default values come from the SignLLM-120M-Base-M (ASL).

of the video, as motion capture models do not exhibit the flickering issue common in generative models, resulting in smaller differences between consecutive frames. While the software can output a higher number of frames for enhanced results, we set the frame rate to 24 frames per second for fair comparisons. In conclusion, we believe that introducing motion capture-related techniques, models, or methods holds great promise in the final rendering stage of sign language.

D.4 Model Parameter Study

As shown in Table 4, we have investigated the optimal parameter settings under different circumstances to provide further discussion and guide future researchers in their training. This includes the optimal results of our primary model parameters, architecture, and various learning rates or other parameters. The experimental results were derived by evaluating the performance of the Text to Pose function from the SignLLM-40M-Base model to the SignLLM-1B-Large model on the ASL part of Prompt2Sign dataset. In general, we find that (1) The optimal values of the parameters conform to the scaling law [34, 40], according to which we should increase the number of parameters by four times when the data is increased by four times. There is no significant difference between the 120M model and the 40M model prediction without too much increase in data volume, and there is also a larger magnitude. (2) When we use Prompt2LangGloss, our data equals the sum of two single language versions. But at this time, their performance mainly depends on the data of a single language, which is a special case: Although they share parameters, the LangGloss has distinguished enough of the sign language pose corresponding to the input text, they do not enjoy the bonus of shared parameters.

[4] LR is getting smaller and smaller over time, approaching a set value.

E Discussions

Discussion on Dataset Selection. We have cited the sources of our publicly available data, and some of the more popular works were not considered due to their limited accessibility and potential usage restrictions. Additionally, while there are other multilingual datasets available, they may not possess the same level of comprehensiveness as ours. Like [30] and [101], they are papers that translate two types of sign language videos into spoken language (SLT), while our work is from spoken language to videos (SLP). We aim to be the first multilingual SLP method, and our dataset has more diverse application scenarios than them (*e.g.*, one is just Bible translation[5] [30], the other[6] [101] is cross SLT, but we are comprehensive scenarios and SLP). We are a very beneficial supplement to previous work, which is quite different from previous work.
Discussion on Dataset Errors. We handle issues related to NaN, zero, and missing data by applying deletion or replacement techniques and our tool simplifies certain calibration stages in comparison to previous 2D to 3D tools, which may introduce some errors. The substituted data is derived using median or mean values, resulting in minuscule errors. Within the vast dataset, these errors typically fall within the range of 0.5% to 0.7% (We conducted a random sampling of results and obtained a ratio of 87 out of 17,549 to 47 out of 6,685). Moreover, our processing steps involving normalization greatly diminish such errors. Hence, we have reasonable grounds to assert that the data error is minimal enough to be bearable. In addition, we consider reducing these errors before release.
Does Prompt2Pose need to be independent as a task? If existing methods want to use long text prompts as input, they must obtain more data to achieve better results. However, sign language data is very scarce. We need to propose a new Prompt2LangGloss method that can efficiently translate based on the scarcity of sign language. Therefore, it is necessary to propose Prompt2Pose as an independent task, as this module has many application scenarios even in a wider range of action synthesis (*e.g.*, the robot can understand human commands more efficiently, so that the GPT4 to summarize the Prompt is not required). Compared to using GPT4 to simplify and summarize everyone's input, creating an efficient end-to-end sign language model is very valuable, similar to the popular Retrieval Augmented Generation (RAG), but for sign language.
Discussion on Similar Name. We noticed that a work [27] has introduced a large language model to translate sign language videos into spoken text. Their proposed framework is also called SignLLM. We are from spoken text to sign language videos, and it is obvious that our work is completely different. We should have left traces online earlier than them, but because our work is completely different, there is no conflict.
Limitations. Although our tool has significantly advanced the automation of sign language data processing and data acquisition, it is not a complete end-to-end solution. For instance, before inputting into our pipeline, batch processing of

[5] https://aclanthology.org/2023.findings-emnlp.664/
[6] https://ieeexplore.ieee.org/document/9878501

sign language videos is required using OpenPose. This batch processing step for sign language videos necessitates some manual work to adapt to the format of the new dataset. Additionally, separating audio and video/clipping videos/aligning transcripts needs to be completed by oneself. Due to the need for researchers to search for relevant sign language data online when obtaining more data. Unmarked videos may have many useless segments, which may potentially lead to some problems, so data quality needs to be improved by the researchers themselves. Finally, due to the learning cost of developing a new format based on previous work, it is currently urgent to develop a universal data loading and reader. Therefore, it is necessary to adapt the model before using Prompt2Sign.