LLM-based Sign Language Production

Wellington Silveira
Center of Computational Sciences
Federal University of Rio Grande
Rio Grande, Brazil
wellingtonfs@furg.br

Luca Mendonça
Center of Computational Sciences
Federal University of Rio Grande
Rio Grande, Brazil
lucabarbozamendonca@furg.br

Rodrigo de Bem
Center of Computational Sciences
Federal University of Rio Grande
Rio Grande, Brazil
rodrigobem@furg.br

2024 International Conference on Machine Learning and Applications (ICMLA) | 979-8-3503-7488-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICMLA61862.2024.00260

***Abstract*—Sign language is an effective means of communication for individuals with different degrees of hearing impairment. Assistive technologies are a significant ally in the social inclusion of people from these groups. Recent advancements in natural language models and high-definition image generation emerged as a novel, non-intrusive approach for tasks related to sign language recognition (SLR), translation (SLT), and production (SLP). This work introduces a novel sign language production approach based on Large Language Models (LLMs) capable of translating text to signs, which are then synthesized into sequences of images. Our method is tested with two LLMs, the LLaMA-7B-hf (quantized) and the Vicuna-7B. Experiments performed on two sign language datasets, the RWTH-PHOENIX-Weather, and the SynLibras-Pose, have shown promising results for this cross-modal application of LLMs.**

***Index Terms*—sign language production, large language models, image synthesis**

I. Introduction

The World Health Organization (WHO) estimates that the number of people with some hearing loss was 430 million in 2021 [1]. One of the ways to mitigate the communication obstacles for people with hearing impairment is using sign language. However, the number of proficient sign language speakers is still small, making communication difficult. In this scenario, the use of assistive technologies, such as those based on computer vision, becomes a significant ally for the social inclusion of these individuals in everyday activities.

Computer vision approaches are non-intrusive methods to perform tasks of recognition, translation, and production of sign language in images and videos. In the latest years, deep generative models, such as GANs [2], VAEs [3], Transformers [4], and Diffusion models [5], have enabled various tasks in the image domain, such as image-to-image translation, recognition, and synthesis. Such tasks have also been explored for sign language production [6]–[8].

More recently, models based on the Transformer architecture have gained notoriety for enabling groundbreaking advances in natural language processing. These models are known as large language models (LLMs). LLMs were initially employed for text-to-text tasks, such as question-answering (QA) and summarization. However, new applications started to be explored in the literature, involving cross-modal processing such as text-to-image and text-to-motion tasks. Notably, LLM-based sign language production is not well explored in the literature yet. Therefore, we introduce a novel approach for SLP, in which a text input is directly translated into motion sequences by an LLM. These motion sequences are subsequently employed to synthesize of subjects performing sign language. Our method is tested with two LLM models, the LLaMA-7B-hf (quantized) and the Vicuna-7B. The experiments performed on two sign language datasets, the RWTH-PHOENIX-Weather, and the SynLibras-Pose, have shown promising results for this cross-modal application of LLMs. Thus, the contributions of this work are summarized as follows: i) The novel use and fine-tuning of pre-trained LLMs to map natural language text to a sequence of sign language poses; ii) A conditional deep generative model to perform pose-transfer, given the appearance of a specific person.

II. Related Works

Until recently, the models used to tackle mapping between sequences were mostly based on recurrent neural networks (RNNs), LSTMs [9], and GRUs [10], e.g. test2sign [11]. However, these models are computationally expensive and do not perform well with long sequences. Based on this knowledge Vaswani et al. [4] introduce the Transformer architecture, a novel approach to handling long sequences while using less computational resources.

Transformer models utilize an encoder-decoder-based architecture which, unlike RNN-based models, does not make use of a hidden state that has to be computed for each element of a sequence to store context. Instead, it assigns degrees of importance (attention) to each element. The encoder, through an attention mechanism, can learn the relationship between words and the importance of each one to the sequence's context. This is expressed through an attention matrix that relates each word to every other word within the sequence.

In recent years, Large Language Models (LLMs), based on Transformers, gained notoriety for their outstanding performance in understanding and processing natural language [12]. In this segment, various models have been developed, e.g. BERT [13], GPT [14], GPT-4 [15], and Google T5 [16]. These models contain an expressive amount of trainable parameters and are trained with large quantities of text data.

SLP is a task focused on generating sign language from spoken language, that is, it translates sentences from spoken language to sign language sentences [17]. We can divide SLP into two distinct subtasks: 1) Text-to-pose translation, that is, given a sentence in spoken language text, generate a sequence

1946-0759/24/$31.00 ©2024 IEEE
DOI 10.1109/ICMLA61862.2024.00260

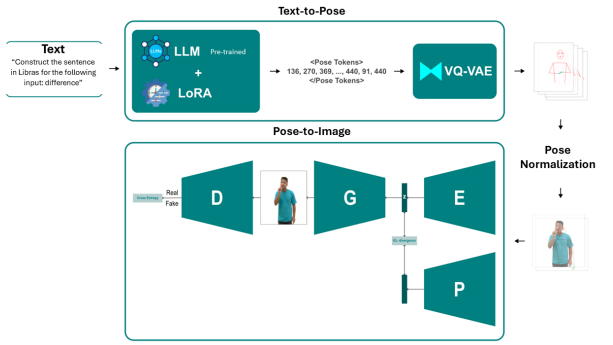

Fig. 1. Proposed Architecture. The text-to-pose mapping and the pose-to-image pose-transfer modules work together to perform text-to-sign language production.

of poses in sign language; 2) Synthesizing a sequence of images (video) from a sequence of poses (pose-to-image).

Some works in the literature perform only text-to-pose, e.g. [7], [18]. However, methods based only on pose synthesis as the final result of translation do not guarantee good understanding by people with hearing impairments when compared to methods for generating images of people speaking these languages [19]. Therefore, the synthesis of realistic images of people speaking in sign language is something desired and increasingly explored in the literature.

Historically, SLP relied on animated graphic avatars [20]–[23]. However, these approaches use pose movement based on rules that do not generalize to unseen sequences [8]. In addition, with the increasing advancement of deep generative models, new ways of performing image synthesis have emerged, serving as an alternative to traditional graphic avatars. In contrast with previous methods, our approach directly employs pre-trained LLM models for sign pose generation, leveraging its capabilities. Moreover, we synthesize images of given signers (appearances) conditioned to body poses, performing pose-transfer. Our method is related to [24], however, despite the fact it does not generate sign language, the related method also does not execute the pose-to-image step, producing only poses as results.

III. Methodology

Our methodology, illustrated in Figure 1, is divided into two main parts: 1) the mapping between spoken language texts and sign language poses (text-to-pose); 2) the mapping between sign language poses and images (pose-to-image), performed as a conditional pose-transfer to given appearances of subject signers.

A. Text-to-Pose

The task of translating text to pose in SLP is usually achieved through an intermediate notation, the glosses. Thus, the text is first transformed into a sequence of glosses that are then mapped to a sequence of corresponding poses [8]. However, through the fine-tuning of an LLM, it is possible to perform this text-to-pose mapping directly through the use of *tokens* that represent these poses [24].

1) VQ-VAE Tokenization: To perform text-to-pose mapping, it is necessary to transform the poses into *tokens*, and to do this the VQ-VAE [25] was used. The fundamental difference between VQ-VAE and traditional VAE is that the former has a discrete latent vector instead of a continuous one. Thus, the VQ-VAE was used to create the *tokens* that represent the sign language poses. The pose keypoints were encoded in a discrete representation to be used in the LLM fine-tuning. This discrete representation (*pose tokens*) can then be decoded by the VQ-VAE *decoder*, returning them to pose points. Figure 2 exemplifies the dynamics of this pose encoding and decoding in the VQ-VAE architecture.

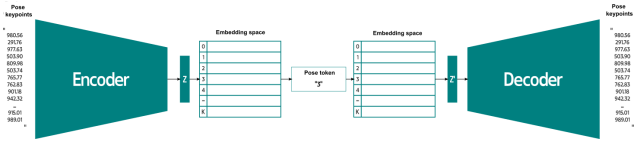

Fig. 2. VQ-VAE Architecture: The keypoints extracted using the human pose estimator OpenPose [26] are encoded by the *encoder* generating the latent vector z. This generated vector z is compared with all the vectors in the embedding space and the index belonging to the vector closest to z is used as *token* for the LLM input.

2) LLM and Dataset Preparation: There are several LLMs with different numbers of parameters in the literature. Among the best-known open-source language models are the models developed by Meta, called LLaMA [27] and the more recent LLaMA2 [28]. Among these models, there are variations regarding the number of parameters. It is specified in the suffix of the model's name, as in LLaMA2-7B, where the letter *B* specifies how many billion parameters the model has.

Another well-known LLM model is the Vicuna. It is a fine-tuned version of LLaMA trained and validated in the FastChat platform [29]. Vicuna was trained with thousands of human conversations, thus, Vicuna is an LLaMA model fine-tuned for chatting. It presents the same LLaMA variations regarding the number of parameters (i.e., 7B, 13B, etc.). For this work, the FastChat platform was used to perform fine-tuning and validation of the LLaMA-7B-hf (quantized) and Vicuna-7B models and to compare their results.

In addition, for an LLM to be fine-tuned, it needs several examples of conversations that deal with the target task. Also in the case of text-to-pose, several variations of texts to perform the mapping are required. Because of this, several versions of input text were created for the same poses, aiming for a sufficiently diverse dataset. Each input text in this dataset specifies the generation of one or more words, in addition to each entry being associated with its respective expected response. Finally, the dataset was created under the FastChat standard to be compatible with the platform.

3) Fine-tuning with LoRA: We employ Low-Rank Adaptation (LoRA) [30], which allows us to keep the original weights of the LLM unchanged. In addition, LoRA provides faster training compared to traditional fine-tuning techniques.

LoRA has some variations regarding its optimization, one of which concerns the quantization of model parameters to reduce storage size. This variation, called Quantized LoRA (QLoRA), uses a quantized model to perform inference, that is, the original model loses part of its floating point precision, going from the traditional 32 bits (FP32) to a lower precision, which can reach 4 bits (FP4). In our case, the 16-bit model, that is, FP16, was used.

B. Pose-to-Image

The second subtask of the SLP is pose-to-image mapping, done here through pose-transfer to given appearances of signers. For this purpose, a Conditional VAE-GAN architecture was used for image generation, closely following [31].

1) Conditional VAE-GAN: Conditional VAEs are already explored in the literature to condition the generation of images [32], [33]. In traditional VAE, the image is encoded in a latent vector that is decoded back into the original image. In conditional VAEs, a new conditioning variable y is introduced to the marginal log-likelihood, as per,

$$\begin{aligned} \log\, p_\theta(x|y) \geq \mathbb{E}_{q_\phi(z|x,y)}\left[\log p_\theta(x|z,y)\right] \\ - KL\left[q_\phi(z|x,y)||p(z|y)\right]. \end{aligned} \tag{1}$$

where, $\log\, p_\theta(x|y)$ is the conditional marginal log-likelihood and the right-hand side of the inequality is the evidence lower bound (ELBO). The two terms of the ELBO are learned simultaneously through an encoder-decoder architecture. In such an architecture the encoder, with parameters ϕ, minimizes the KL-divergence between the surrogate distribution $q_\phi(z|x,y)$ and the prior distribution $p_\theta(z|y)$, while the decoder, with parameters θ, minimizes the expected L1 reconstruction loss of the generative model $p_\theta(x|z,y)$. In our model, y represents a conditioning pose, transferred by the model to a given image with the appearance of a signer subject.

To improve the quality of the generated images, a GAN discriminator with a cross-entropy loss is added to classify the images generated by the decoder as real or fake, turning the final model into a Conditional VAE-GAN (CVAE-GAN). In addition, *pixelwise* normalization was added to all layers of the decoder (generator) to stabilize the model training. This normalization was introduced by Karras et al. [34], as well as the equalized learning rate, which was also used in our model. The architecture of the pose-to-image module is illustrated in Figure 1.

IV. Experiments and Results

A. Datasets

Two datasets were used to train and test the models. Firstly, the SynLibras-Pose [31], a Brazilian Sign Language (LIBRAS) dataset containing approximately 1,133 videos from four different signers. Each video includes a sequence of movements corresponding to a Portuguese word or expression in LIBRAS, with approximately 200 frames with 1024×1024 pixels resolution. All videos are annotated frame-by-frame with signer poses estimated with OpenPose [26] and revised manually. Figure 3 shows a sample from the SynLibras-Pose dataset. Secondly, the RWTH-PHOENIX-Weather [35], a German Sign Language dataset. It contains videos with 25 images per second from nine different signers. The resolution of frames is 210×260 pixels. Figure 4 shows a RWTH-PHOENIX-Weather sample.

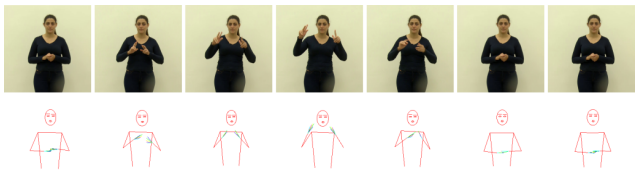

Fig. 3. Sample from SynLibras-Pose. A sequence of images with the estimated poses.

Fig. 4. Sample from RWTH-PHOENIX-Weather.

The SynLibras-Pose dataset was used both for fine-tuning the LLM for text-to-pose mapping and for training the pose-to-image (CVAE-GAN) model. The annotated poses provided in the dataset were used for the LLM fine-tuning. For training the pose-to-image model poses were extracted with Google MediaPipe [36].

Despite the RWTH-PHOENIX-Weather dataset being in German Sign Language, it also can be used in the Conditional VAE-GAN model training, since only appearances and poses are relevant for this task. Neither the sequence of poses nor their meaning are explicitly important to the CVAE-GAN. However, employing the RWTH-PHOENIX-Weather beneficially increases the variety of appearances learned by the model and the amount of training data.

In addition, the poses were normalized before being transferred between different signers. To perform this normalization, a scale factor was extracted considering the distance between the top of the head and the waist of the person. The reference point for the translation was the midpoint between the shoulders. The normalization is defined as

$$Scale = \frac{\Delta Y_{source}}{\Delta Y_{target}}, \tag{2}$$

where, ΔY_{source} and ΔY_{target} are the differences between the top of the head and the waistline from the source pose and the target pose, respectively. MOx and MOy are the x and y coordinates of the midpoint between the shoulders. Figure 5 illustrates the computation of the scale factor.

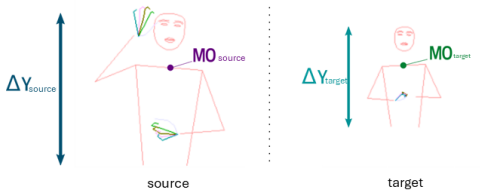

Fig. 5. Points used in pose normalization. Equation 2 shows the calculation performed with the highlighted points. MO is the midpoint between the shoulders and ΔY is the distance from the top of the head to the waist.

V. Text-to-Pose Results

To perform the translation and mapping task between spoken text and sign language poses we fine-tuned the LLaMA-7B and Vicuna-7B LLMs models with LoRA on the FastChat platform comparing their results. In addition, the intermediate results of each model are also compared. Finally, inference tests of individual textual sentences are presented.

1) VQ-VAE Tokenization Results: VQ-VAE was trained to obtain a low-dimensional representation of the poses. In this work, each pose was represented by a single *token* in the embedding space. All poses from SynLibras-Pose were used in the training of VQ-VAE. This model was trained for 45 epochs using PyTorch with the Adam optimizer [37] and the mean squared error (MSE) loss. Figure 6 shows some examples of reconstructions made with the VQ-VAE tokenization.

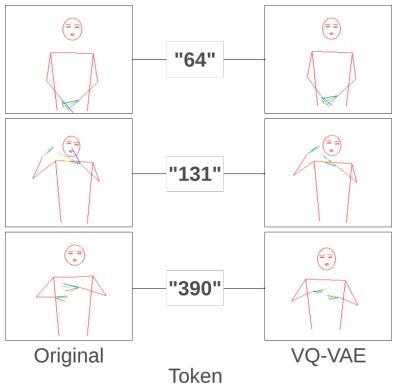

Fig. 6. Reconstruction with VQ-VAE. The first and last columns show the original and reconstructed poses, respectively. In the central column, the *pose tokens* by VQ-VAE.

Figure 7 illustrates an example of encoding a sequence of poses transforming into a sequence of *tokens* that will be used in the dataset for training the LLM.

2) Dataset: The dataset for fine-tuning the LLMs was composed of 170 diverse instructions for each pose sequence. Since each SynLibras-Pose word has a sequence of corresponding poses, this sequence of poses is converted into a sequence of *tokens* with VQ-VAE, which are then placed in the dataset with 170 variations of instructions. In addition, a small validation set was created with three different instructions for a subset of 40 SynLibras-Pose words, with 293 words used in the training. In addition, a standard instruction was used for all entries, as if it were the user's requested prompt.

3) LLM Training: As previously mentioned, we fine-tuned the LLaMA-7B and the Vicuna-7B mode using LoRA [30] as an adapter for the LLM layers. LoRA has some adjustable hyperparameters that control the amount of learnable parameters. The first hyperparameter controls the dimension of the vector r and the second hyperparameter called a serves as a scale factor for the adjustment of the LoRA matrices A and B. At the beginning of the training, the matrix A starts with values sampled from a standard Gaussian distribution, while the matrix B starts at zero, ensuring a more stable training start. The values used for r and a were 32 and 16, respectively, following [24]. In addition, a *dropout* of 0.05 was used, a batch size of 10 with gradient accumulation of 26, i.e., it has an effective batch of 260.

To compare the performance of the models a quantitative score must be used. However, considering that the LLM output is a sequence of numeric *pose tokens* generated by VQ-VAE and that this sequence may be misaligned in time regarding the ground-truth sequence, a direct comparison between them would not be the best option. Therefore, we use Dynamic Time Warping (DTW) [38] to measure the similarity between the generated and ground-truth pose sequences.

Table I shows the results of the DTW technique for the different values of temperature (Temp) and repetition penalty (RP) of the LLaMA-7B model. To use DTW, the *pose tokens* were mapped to pose images with VQ-VAE. Then, the L1 distance between the pose images of the dataset sequences and those generated by LLM was used to populate the cost matrix. Furthermore, Table II shows the percentage of valid sequences generated by LLM. For a sequence to be considered valid, it must follow the dataset response pattern and generate only numeric *pose tokens*. Finally, Table III shows the DTW results for Vicuna-7B and Table IV the percentage of valid sequences for different configurations (RP and Temp values) of this model.

As seen in Tables I, II, III, and IV, defining the best configuration for the models is not trivial, considering that the configurations that lead to the generation of more valid sequences are not the same ones that lead to the best DTW scores. Therefore, the best configuration chosen for the models takes into account a balance between the DTW scores and the percentage of valid sequences. Considering this, the model that achieved the best balance was the Vicuna-7B. With a combination of temperature equal to 0.2 and repetition penalty of 1.2, the model generated 89.5% of valid sequence with a DTW score of 43.170

VI. Pose-to-Image Results

After the first sub-task is obtained through the LLM fine-tuning, it is possible to have the sequence of poses to perform the second sub-task, which is the pose-to-image conditioned pose-transfer. The model to perform this transfer (Conditional VAE-GAN) was trained using the RWTH-PHOENIX-Weather

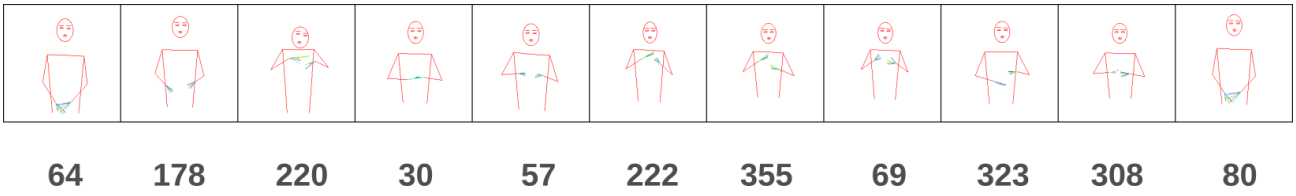

Fig. 7. VQ-VAE encoding. The first row shows samples of a sequence of poses for the word "garlic" in LIBRAS. The second row shows the corresponding *pose tokens* generated by VQ-VAE.

TABLE I
LLAMA-7B RESULTS FOR DIFFERENT REPETITION PENALTY AND TEMPERATURE VALUES. THE SCORE IS BASED ON THE DTW FOR EACH VALID SEQUENCE GENERATED BY THE LLM (SMALLER IS BETTER).

Temp / RP	0.0	0.1	0.2	0.3	0.5	0.7
1.0	43.738	40.619	43.166	42.515	42.758	44.701
1.1	43.057	42.709	42.840	42.847	43.288	42.987
1.2	43.341	43.195	43.075	43.218	44.852	42.275
1.3	43.329	44.597	43.950	42.744	42.976	43.266
1.4	43.920	44.257	43.540	42.646	44.403	43.959
1.5	43.927	43.991	41.482	42.597	43.607	43.243

TABLE II
LLAMA-7B RESULTS FOR DIFFERENT REPETITION PENALTY AND TEMPERATURE VALUES. PERCENTAGE OF VALID SEQUENCES OUT OF 200 GENERATED (LARGER IS BETTER).

Temp / RP	0.0	0.1	0.2	0.3	0.5	0.7
1.0	48.0	41.0	43.0	45.5	47.0	53.0
1.1	64.0	60.5	59.0	60.0	59.0	52.0
1.2	70.0	69.0	65.5	61.0	57.0	54.5
1.3	71.5	72.0	68.5	64.0	60.0	43.5
1.4	68.5	71.0	61.0	59.0	54.0	37.0
1.5	54.5	54.0	52.0	49.0	30.0	20.5

and SynLibras-Pose datasets. The combination of the datasets resulted in approximately 40,000 images resized to a resolution of 256×256 pixels for training. Figure 8 shows images from the dataset with their estimated poses.

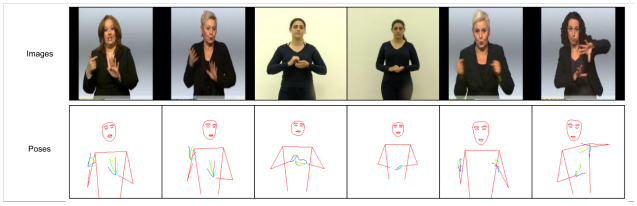

Fig. 8. Datasets samples with their estimated poses.

The model was trained with approximately 13 epochs with a batch size of 14 images using PyTorch with Adam Optimizer. Figure 9 shows the result of the model reconstruction for some test images. Furthermore, Figure 10 shows images generated in the pose-to-image mapping in which poses are transferred to given appearances of signers.

TABLE III
VICUNA-7B RESULTS FOR DIFFERENT REPETITION PENALTY AND TEMPERATURE VALUES. THE SCORE IS BASED ON THE DTW FOR EACH VALID SEQUENCE GENERATED BY THE LLM (SMALLER IS BETTER).

Temp / RP	0.0	0.1	0.2	0.3	0.5	0.7
1.0	42.463	42.506	43.664	43.787	44.930	43.661
1.1	43.775	44.504	43.653	44.661	44.103	44.136
1.2	44.513	43.955	43.170	43.825	44.369	45.203
1.3	44.621	44.122	44.961	43.480	44.554	43.347
1.4	44.697	44.871	45.196	44.538	42.435	44.265
1.5	44.411	45.745	44.804	45.731	42.378	43.186

TABLE IV
VICUNA-7B RESULTS FOR DIFFERENT REPETITION PENALTY AND TEMPERATURE VALUES. PERCENTAGE OF VALID SEQUENCES OUT OF 200 GENERATED (LARGER IS BETTER).

Temp / RP	0.0	0.1	0.2	0.3	0.5	0.7
1.0	50.0	53.0	54.5	60.0	75.0	84.0
1.1	83.0	85.0	89.0	89.0	89.5	85.0
1.2	87.5	88.0	89.5	86.5	83.0	80.0
1.3	82.0	83.5	85.0	81.5	81.5	70.5
1.4	66.5	74.0	60.5	61.5	52.0	38.5
1.5	37.5	36.5	36.5	37.0	30.0	21.5

VII. CONCLUSION

We introduce a novel LLM-based method for sign language production (SLP). The technique is composed of text-to-pose and pose-to-image steps. In the former, a pre-trained LLM is fine-tuned with LoRA methods for generating VQ-VAE tokens from text. Such tokens are subsequently mapped to body poses. In the latter step, the poses are transferred to images of given subjects (signers) by a Conditional VAE-GAN method. Experiments were performed with two pre-trained LLMs, the LLaMA-7B-hf (quantized) and the Vicuna-7B, on two datasets, the SynLibras-Pose, and the RWTH-PHOENIX-Weather. Although with limitations, the LLM fine-tuning with LoRA is capable of generating coherent pose sequences for different created instructions. Similarly, the employed pose-to-image model can also reconstruct input images and perform pose-transfer, yet it produces some artifacts and blurs in some regions (e.g. faces and hands).

ACKNOWLEDGMENT

Luca Mendonça is a PIBIC/CNPq scholarship holder under process 129864/2024-2.

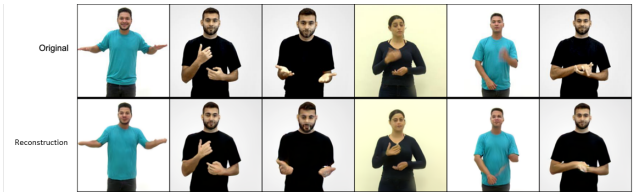

Fig. 9. Reconstruction performed with the Conditional VAE-GAN model.

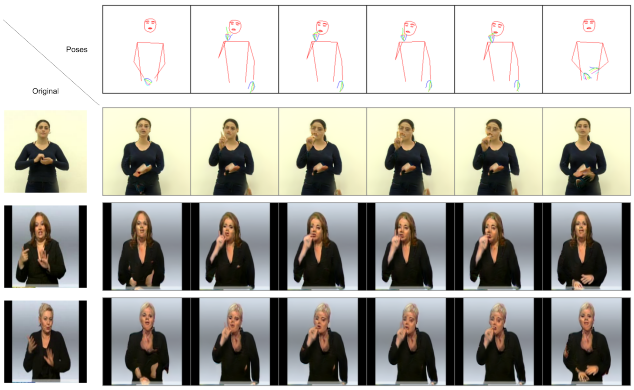

Fig. 10. Pose-to-image results. Poses: The conditioning poses for the word "water" in Libras. Original: Single images of signers with diverse appearances from the test set. Poses are transferred to the given appearances generating a sequence of new synthetic images.

REFERENCES

[1] WHO, "Deafness and hearing loss," https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss, 2021.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NeurIPS*, 2014.

[3] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *ICLR*, 2013.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.

[5] J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *ICML*, 2015.

[6] B. Saunders, N. C. Camgoz, and R. Bowden, "Everybody sign now: Translating spoken language to photo realistic sign language video," *arXiv preprint arXiv:2011.09846*, 2020.

[7] ——, "Progressive transformers for end-to-end sign language production," in *ECCV*, 2020.

[8] ——, "Signing at scale: Learning to co-articulate signs for large-scale photo-realistic sign language production," in *CVPR*, 2022.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[10] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *EMNLP*, 2014.

[11] S. Stoll, N. C. Camgoz, S. Hadfield, and R. Bowden, "Text2sign: towards sign language production using neural machine translation and generative adversarial networks," *IJCV*, vol. 128, pp. 891–908, 2020.

[12] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM Trans. on Intelligent Systems and Technology*, 2024.

[13] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019.

[14] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," OpenAI, 2018.

[15] J. Achiam, S. Adler *et al.*, "Gpt-4 technical report," OpenAI, 2023.

[16] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *JMLR*, 2020.

[17] R. Rastgoo, K. Kiani, S. Escalera, and M. Sabokrou, "Sign language production: A review," in *CVPR Workshops*, 2021.

[18] J. Zelinka and J. Kanis, "Neural sign language synthesis: Words are our glosses," in *CVPR*, 2020.

[19] L. Ventura, A. Duarte, and X. Giró-i Nieto, "Can everybody sign now? exploring sign language video generation from 2d poses," in *SLRTP Workshop*, 2020.

[20] J. Kennaway, J. R. Glauert, and I. Zwitserlood, "Providing signed content on the internet by synthesized animation," *ACM TOCHI*, 2007.

[21] P. Cabral, M. Gonçalves, H. Nicolau, L. Coheur, and R. Santos, "Pe2lgp animator: A tool to animate a portuguese sign language avatar," in *LREC Workshop*, 2020.

[22] R. Elliott, J. R. Glauert, J. Kennaway, I. Marshall, and E. Safar, "Linguistic modelling and language-processing technologies for avatar-based sign language presentation," *Universal access in the information society*, vol. 6, pp. 375–391, 2008.

[23] Hand Talk, https://www.handtalk.me/en, 2021.

[24] Y. Zhang, D. Huang, B. Liu, S. Tang, Y. Lu, L. Chen, L. Bai, Q. Chu, N. Yu, and W. Ouyang, "Motiongpt: Finetuned llms are general-purpose motion generators," in *AAAI*, 2024.

[25] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *NeurIPS*, 2017.

[26] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE TPAMI*, 2019.

[27] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[28] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[29] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing *et al.*, "Judging LLM-as-a-judge with MT-Bench and Chatbot Arena," in *NeurIPS*, 2023.

[30] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *ICLR*, 2022.

[31] W. Silveira, A. Alaniz, M. Hurtado, B. C. da Silva, and R. de Bem, "Synlibras: A disentangled deep generative model for brazilian sign language synthesis," in *SIBGRAPI*, 2022.

[32] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *NeurIPS*, 2015.

[33] R. de Bem, A. Ghosh, T. Ajanthan, O. Miksik, A. Boukhayma, N. Siddharth, and P. Torr, "Dgpose: Deep generative models for human body analysis," *IJCV*, 2020.

[34] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[35] J. Forster, C. Schmidt, O. Koller, M. Bellgardt, and H. Ney, "Extensions of the sign language recognition and translation corpus rwth-phoenix-weather." in *LREC*, 2014.

[36] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann, "Mediapipe: A framework for perceiving and processing reality," in *CVPR Workshops*, 2019.

[37] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic gradient descent," in *ICLR*, 2015.

[38] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.