UTRECHT UNIVERSITY

Department of Information and Computing Science

Game and media technology master thesis

Co-speech gesture generation with non-deterministic deep learning methods

Thesis supervisor:

Prof. Dr. Albert Ali Salah

Second examiner:

Prof. Dr. Remco C. Veltkamp

Candidate:

Soheil Bagheri

Student number:

6208908

August 14, 2024

Abstract

Gestures are the non-verbal part of the communication that accompanies speech to convey meaning to the audience. By this means gesture generation has been an open topic for research for several years. In recent years, with the advancements in deep learning algorithms and the development of larger datasets, generating natural human gestures enhanced significantly. However, because of the idiosyncratic, non-periodic, and diversity of human motions, still generated gestures suffer from the semantic relevancy with the input speech. In this thesis, we will focus on the semantic grounding problem and develop non-deterministic machine learning methods that can also tackle this problem. The method is expected to generate semantic motions for both arms, fingers, and torso. In the end, the method will be evaluated both objectively and subjectively with respect to ground truth and state-of-the-art methods.

Contents

1. Introduction

1.1 Introduction

David McNeill [1] describes gestures as symbols that have been used to show meaning with hands. They can be used to emphasize the spoken context [2], make them more clear [3], or even enhance the expression of that information [4][5]. Further researches show that accompanying gestures with speech can increase the persuasion, credibility, and concentration of the audience [6][3]. Other than that, gestures can show understanding when humans are talking to non-human characters [7]. So, we can conclude gestures are essential to have a natural and lively speech [4][2] and because of that researchers started to synthesize gestures for non-human characters.

Synthesized gestures usually have been used for digital characters [4][5], humanoid robots [5], or human-machine interaction [3]. They can be used in animation [4], film [2][4], game industry [5][2], or even for educational purposes [4]. Their ability to enhance communication can benefit all of these businesses. However, these synthesized gestures should be natural and aligned with the context to affect the audience properly.

When it comes to generating gestures from speech that are semantically aligned with each other different challenges come along the way. For example, there is a lack of a large dataset consisting of a wide range of gestures from different cultures [8][9]. This is because creating an accurate motion-captured dataset requires a lot of investment and human labor and pose estimation from videos is not highly accurate yet [10]. In addition, while capturing these gestures, everyone has some random movements that add some kind of noise to the actual gesture data [5].On the other hand, there is a large variety of gestures that need to be processed and generated which makes it hard to use either rule-based or data-driven approaches [5].

Also, still, the nature of gestures is ambiguous [6] and it is even hard to find a pattern for them intuitively. The other hurdle is that the dynamic pattern of different human parts is not the same [3] and different variances of their movement make the training more complex. Furthermore, human gestures can be influenced by various factors that cannot be extracted from only speech [6]. For example, the environment that the speech is given and the objects that the speaker is pointing at. In addition, different people use different gestures to describe the same phenomena and it makes it harder for models to find a general pattern for everyone [6][10].

Finally, semantic cues are hard to find because their nature is still ambiguous, and gestures and speech are not temporarily aligned well [4]. Also, beat cues are easier to find because of the alignment of the frequency of the audio and body movement, so deep-learning models are more inclined to learn from them, and less attention will be on semantic cues [4].

In this research, we try to apply contrastive learning to extract underlying semantic alignment between the context or transcript of the speeches and the gestures. Also, because of the idiosyncratic nature of the gestures we combine the speaker's identity with contrastive learning to learn the speaker's gesture style rather than giving it as an input to the model. Moreover, we explore non-deterministic approaches such as diffusion models as they showed promising results in the other generative fields, like image, video, mesh, and even general motion generation.

2. Related Works

In this section, we briefly explain the definition of gesture and its kinds through the lens of linguistics 2.1, then we explore the history of gesture generation and the techniques and algorithms that were acquired 2.2, followed by a deep dive into deep learning algorithms and their usage in co-speech gesture generation 2.3 and the main challenges and concerns 2.4. Further, we discuss some of the key elements like encoders and datasets that have been used previously in this line of research 2.5.

Nyatsanga et al. [11] provided a survey about gesture generation that covers most of the history of this topic. They categorized the previous approaches, based on the input they have used, like text, audio, text and audio, and other modalities like speaker identity and personality [12]. We also recommend reading that to understand the importance of different modalities in this topic.

However, like many other research, we categorize them based on the algorithms they have used. From the higher view, we can categorize gesture generation algorithms into two main categories, rule-based methods, and data-driven methods, where the data-driven approach itself can be seen as statistical, motion graph, and machine learning approaches.

In the following, we define each category and some of their sub-categories and discuss some of the examples that used that algorithm for gesture generation. Later, we explain the nature of the gestures, and the mutual concerns we observed in these examples, and we will finish this section with a survey on some of the datasets and tools that are used in this research area.

2.1 Gestures

McNeill [1] says during the speech, the hands of the speaker are not hands moving in the space anymore. But, they are symbols that come with the speech to convey meanings. Then, he proposes four types of gestures, iconic, metaphoric, deictic, and beat. Iconic gestures depict concrete objects while metaphoric gestures are the ones that show abstract things. Deictic gestures are the ones we use to point at something, and beat gestures are the ones that come with the rhythm of the speech.

Neff et al. [13] propose two new categories called self-adaptors for the gestures that depict the emotional or personality traits of the speaker, and emblems which are conventional gestures that represent a word [14].

Beat gestures usually accompany the speech and because of their alignment with the audio the generation of them is easy and already solved by different methods. However, the other types of gestures, which we call semantically aligned gestures, are more sparse and contextual making them harder to be extracted from the speech and generated by deep learning models. As a result, generating semantically aligned gestures is still an open topic to research, and in this research, we are going to focus on improving their generation.

2.2 History of gesture generation

In this section, we explain the history of gesture generation and the algorithms that have been used for this purpose. We start with rule-based algorithms that have a longer history and describe how gesture generation evolves with new advances in computer science and deep learning. A brief description of each algorithm and some examples from each of them have been provided.

2.2.1 Rule-based approach

In the rule-based approach, an expert models different scenarios and mappings between speech and related gestures. This approach needs human labor, and the transition between gestures can be unnatural as usually the end states are modeled and transitions are calculated naively. The number of variations is limited, and it is hard to scale because each of the variations should be generated and mapped manually which needs not only human labor but also some experience. However, it is really communicative and usually semantically relevant in a limited domain because they are specifically designed to behave in a certain manner [11][15][12].

The rule-based approach can be seen as a classification task to choose the optimum pre-recorded motion from a database, while the early data-driven approaches tend to generate motions as a regression and later probabilistic end-to-end task [16][17]. One of the earliest works using this method was done by Cassell et al. [18] in 1998. They introduced a framework to generate context-related hand gestures, facial animation, and intonation between humanoid characters.

2.2.2 Data-driven approach

In contrast to the rule-based approach that relies on pre-defined rules, the data-driven approach finds its way through data and the patterns that can be found among that. We can categorize the data-driven approach into some sub-categories like statistical, motion graph, and deep learning approach. In the following, we discuss each of these sub-categories briefly, and then in section 2.3 we dive deeper into the deep learning approach which is the state-of-the-art method.

2.2.2.1 Statistical approach

In comparison to the rule-based approach, where an expert connects gestures to speech, in the statistical approach some computation is done to find a direct or probabilistic relation between the speech and the occurrence of the gesture [12].

One of the earliest works on this approach has been done by Kipp [14] who made a profile for an individual from the annotated co-speech dataset. His model was able to generate gestures for this person based on their profile. This profile is about the conditional probability of the occurrence of a gesture along the input text [11].

This approach can be scaled better than the rule-based approach as statistical analysis can be done on different data and generate different results. However, in comparison to the deep learning approach, it is more dependent on the human and their ability to analyze the data.

2.2.2.2 Motion graph

Although deep neural networks can find some deeper connections between the audio, transcript, and gestures, using a low dimensional latent space made them less sensitive to high-frequency motion details. On the other hand, motion graph algorithms have shown higher fidelity in generating these kinds of motions.

The core idea of the motion graph is to generate a graph of the mapping of gestures and audio in a high-dimensional space where each node represents a motion clip and each edge represents the cost of transition from each node to the adjacent one. The best gesture at each sequence can be found by evaluating these cost values with different algorithms like KNN [19] or reinforcement learning [20] in this space [21].

2.2.2.3 Deep learning algorithm

After showing great results in other generative domains, deep learning algorithms were acquired in gesture generation. In comparison to previous approaches, deep learning algorithms need more data to be trained on. However, they need less human effort to process and find the underlying patterns [6].

Kopp [22] argues that using deep learning algorithms has improved the generated results regarding naturalness. However, the communicative part or the semantic connection between speech and gestures might not be as good as the previous approaches [11].

Early attempts led to natural gestures that were aligned with the rhythm of the speech however the semantic relation was not pretty well[2]. One of the reasons is that for deep learning models finding rhythmic cues from audio is easier than finding the semantic cues and the models are more inclined to learn from them [4]. The inclination is because of the distribution of these cues. While rhythmic cues are distributed all over the speech, semantic cues are more sparse. So, because the semantic cues do not follow a clear pattern, they are harder to find by deep learning models, and sometimes they are considered noise.

Later, the focus moved to the semantics of the generated gestures which is harder and still an open topic to research [23][8]. In this research, we focus on enhancing the semantics of the synthesized gestures by using deep learning models.

2.3 Deep learning algorithms

Different deep-learning algorithms have been developed for generating purposes which are also used by researchers for generating co-speech gestures. Sometimes these researchers focused on a specific algorithm, and sometimes they combined some of them. Here is a list of some of the main generative algorithms and the papers that used them:

- RNN [12]
- LSTM [24][25][10][26]
- GAN [6][27][28]
- Autoencoders [29][30][31][32][26]
- Transformers [33][27][8][34]
- Diffusion models [16][35][5][7][36][37]

2.3.1 RNN

Recurrent neural networks [38] are some kinds of neural networks that have been designed for modeling sequential data. They have a loop in their architecture that helps the model at each sequence to use the memory of the previous sequences and find a pattern in sequential data. However, the traditional RNNs suffer from the problem of vanishing the gradient where the gradient becomes so small in long-term dependencies and makes the training difficult for the model.

Ferstl et al. [12] tried to generate gestures from speech using an RNN algorithm that was designed to find patterns in human gesture sequences. They proposed that transfer learning can be helpful in this case because of the complexity that comes with the multi-modality nature of the problem. So, they trained a model on the gesture-to-gesture dataset to find a pattern between motion sequences first and then used that model for training a speech-to-gesture model. However, the general motion joint movement distribution is completely different than gestures, and motion movements are not necessarily linked with speech.

2.3.2 LSTM

LSTM [39] is one kind of recurrent network that came to address the vanishing gradient problem in the traditional one. In LSTM architecture each sequence has three gates that decide to remember or forget the previous memory from the previous sequences and have a constant error through the time. These features help LSTM to find patterns in longer sequential data. Although LSTM has been designed to model sequential data, its deterministic nature makes it not suitable for modeling human gestures which is non-deterministic.

Shlizerman et al. [40] tried to find the correlation between the body joints of a pianist or violinist and music using the LSTM algorithm. Later, Wang et al. [25] used the LSTM algorithm for generating both the speech and gesture from the text input. However, generating rhythmic aligned gestures without using the audio can be challenging and very dependent on the text-to-voice algorithm.

Liu et al. [10] developed a multi-modal LSTM network that uses audio, text, emotions, speaker identity, and facial blend weights as inputs to synthesize gestures. Their model showed good results for that time, however, the deterministic nature of the LSTM algorithms prevents them from generating diverse gestures.

2.3.3 GAN

Generative adversarial networks are a framework consisting of two models that are trained together. The goal of the first model is to generate data from the inputs and the goal of the second model is to predict whether the given input is real or fake [41].

Because of the coherency between the gesture and speech content, speech audio, personality, and environment, Yoon et al. [6] designed an end-to-end co-speech gesture generation model that uses GAN architecture and audio, text, and speaker identity as input to generate more expressive gesture motions. Their model was the first of the kind that used these three modalities at the same time and showed the importance of each of these modalities in gesture generation. Inspired by them, we also use these modalities with some tweaks in interpreting them.

Habibie et al. [28] first tried to generate the facial and gesture motion at the same time using an adversarial network. But, new research [34] showed that generating hand and other body parts separately can enhance the results, because of the different distribution of their joints. So, combining face and body generation might not be the best idea.

All in all, GAN models are hard to train, because they suffer from mode collapse, and they might not be generalized to produce good results for unseen data [5][7]. Mode collapse is assigned to the GAN models when the generator focuses on a subset of data and fails to capture the full diversity of the dataset. While the discriminator increasingly learns to distinguish the real and fake data. As a result, the generator is more inclined to generate the samples that have already been learned and neglect the ones that have already been forgotten.

2.3.4 Autoencoders

Autoencoders are some type of neural networks that consist of encoders that encode the input into a compressed latent space, and decoders that can generate the outcome from the latent space. Autoencoders can be very useful in dimensionality reduction, feature extraction, and noise reduction [42]. These abilities convinced us to instead of working on raw motion data, train a type of autoencoder first, and work on the latent version of the gesture motions.

Kucherenko et al. [29] used this technique to generate gestures from speech. First, they trained an encoder to encode the motions into a latent space, and a decoder to generate gestures from the latent space. Then, they trained another encoder to encode speech into the same latent space, so they can generate gestures from them using the same decoder. They also evaluated different lengths for latent space and the use of different features of speech for co-speech gesture generation. They found out that mel-frequency cepstral coefficients perform the best for Co-speech generation which later has been used in different researches in this topic.

Taylor et al. [31] explored the use of autoencoders along with normalized flow to generate gestures for both speaker and listener. Normalizing flow has a better capacity to estimate the latent space than regular Gaussian.

Li et al. [30] proposed separating the latent space into shared code and motion-specific code can be beneficial because of the one-to-many nature of the gestures. However, using only audio can suffer from extracting the semantic cues.

Yi et al. [43] attempted to generate facial expressions and gestures using VQ-VAE. [44] VQ-VAE is a type of autoencoder that instead of encoding data into a continuous space, uses a quantized codebook and instead of a pre-defined distribution of data, learns it through the training.

Because the VQ-VAE models are more interpretable than VAE models and are less likely to suffer from posterior collapse, we decided to use a VQ-VAE model to convert gesture motion sequences into latent space. Posterior collapse happens when the decoder ignores some parts of latent variables are mostly relies on the other parts. This problem can decrease the effectiveness of the auto-encoder and its efficiency.

2.3.5 Transformers

Transformer [45] architecture was invented to deal with low-performance and temporal dependency of the recurrent neural networks. In this architecture, using the attention mechanism, sequential data can be processed in parallel. In addition, by using multi-head attention, the relation between each sequence and further sequences can be found more easily.

Ahuja et al. [33] pointed to the relationship between the sub-words and the gesture. They wanted to know how different intonations for a specific text can change the gesture. Thus, they used transformers to find the relations between sub-words in language and acoustics.

Because of the cross-modality nature of the gesture synthesis, Pang et al. [8] came up with the idea of using transformer encoder-decoder with early and intermediate modality fusion. Also, they acquired an intra-modal pre-training strategy to deal with the limited data available for training on this topic.

After the invention of transformers by Google, OpenAI introduced Generative pre-trained transforms [46] that are the basis of current large language models. The authors showed that pre-training on large un-labeled data using transformers and fine-tuning that later for task-specific problems can improve the outcomes. Zhang et al. [34] very recently used this technique for semantically aligned gesture generation.

Using a GPT-based algorithm they generate rhythm-aware gestures. Then, they retrieve related semantically aligned gestures from a pre-defined dataset and combine it with the rhythm-aligned gesture. They made a dataset containing the most common semantic gestures for this purpose. This model showed state-of-the-art results regarding semantic alignment. However, LLM models are pretty large and hard to train. In addition, their retrieving approach is deterministic and hard to scale.

2.3.6 Diffusion models

Diffusion models [47] have shown high-quality and diverse results in other generative domains, like image generation [48] and video generation [49]. These models consist of two phases where in the first phase or forward phase, the model gradually adds noise to the data until the data turns into a Gaussian noise. In the second phase, the model learns to do the process in reverse and turns the Gaussian noise into the original samples. By learning these transitions the model will be able to generate new results based on different distributions of the data.

Alexanderson et al. [16] made one of the first attempts to use audio input with diffusion models for generating gesture and dance motions. They used conformers [50] to extract the local and global features from the audio. Conformers are a mixture of transformers that are good for extracting global interactions and CNNs that are suitable for extracting local features. Conformers showed better results than transformers and CNNs individually for the speech recognition tasks. In addition, they used classifier-free guidance to control the intensity of the style they wanted to apply. Furthermore, they did research on using products of expert diffusion models and their use case in motion generation. Relying only on audio made their results good in rhythm alignment but poor in semantics.

Ao et al. [35] proposed a diffusion-based method that instead of using limited style embedding from labels or using single modality like sample motions, uses a CLIP-guided [51] mechanism to extract the style from different modalities such as text, video, and motions to generate stylized gestures. Also, they argue that because of the ambiguous and many-to-many nature of gestures, synthesized results lead to mean gesture motions. So, they used contrastive learning and a temporal aggregation mechanism on a latent space of gesture-transcript and body-joint embedding to avoid mean gesture motions and generate semantically aligned gestures. Furthermore, they proposed a self-supervising approach to alleviate the limited labeled data in this field. This approach might be the closest approach to our method, with the difference that we use contrastive learning on audio and style as well as the transcripts.

Zhi et al. [23] came up with a two-step gesture generation idea consisting of a Semantic-Aligned Gesture generator and a Rhythm-Aligned Gesture generator. The SAG part includes an auto-encoder that is trained with two losses. The reconstruction loss calculates the distance between ground truth and the generated gesture, and a CLIP loss measures the distance between the encoded gesture sequence and encoded transcripts using a pre-trained CLIP model. They also used an N-layer MLP network instead of a simple denoiser in their diffusion architecture. They believe that this idea can enhance the quality of the outputs regarding temporal consistency. While analyzing the code of this paper, we realized that for contrastive learning they only considered the cosine similarity of two sets, while it has been proved that considering positive and negative samples can improve the outcome of contrastive similarity loss. [52].

In contrast to previous research that used static data and conditions, and was guided by pre-trained models and CLIP, Zhu et al. [7] proposed a more challenging method called DiffGesture, where both gesture and audio are sequential and audio-gesture mapping is implicit. They used an audio-gesture transformer to find a correlation between speech and gesture and proposed a Diffusion Gesture Stabilizer to solve the temporal distance between audio and motion. Because of the random noise in the diffusion models, temporal consistency is an issue in these models. Zhu et al. [7] by annealing the noise tried to control this randomness to get more consistent results. However, this variance control can result in less diverse outcomes.

Deichler et al. [17] also acquired CLIP guidance to find a semantic between motion, text, and audio. Although they achieved the highest human likeness and speech appropriateness rating at the GENEA conference in 2023 we believe using the concatenation of audio and transcript embeddings in contrastive learning might not be a good idea as their distribution is completely different. So, in our model, we implemented three separate contrastive losses for getting the similarity of gesture and audio, transcript, and style.

Chen et al. [36] tried to generate facial expressions and gestures at the same time to keep the coherency between them. Their approach can be considered as a holistic approach that generates body and face at the same time. We considered this paper for comparison to see if our discrete generative model that generates hand and other body parts separately can outperform these holistic approaches or not.

Although diffusion models have shown high-quality and diverse results in this field still generated results are not aligned semantically well with the context of the speech. And, although training of diffusion models is more stable than the GAN models, these kinds of models are computationally heavy, and training them or getting inferences from them needs more computational power in comparison to the other generative methods, like GANs and VAEs.

2.4 Concerns and Challenges

2.4.1 Type of input

Deep learning algorithms for generating co-speech gestures can be categorized by the type of inputs they use to generate the appropriate gestures [11]. Generally, they use audio, text, or some other non-communicative modalities like the personality, emotion, or face of the speaker [3] or sometimes a combination of some of them to generate gestures.

Early models focused on the audio and got rhythmic aligned results from the prosody of the speech. However, they were not semantically aligned because of the limitation in the information extracted from the audio. Some others focused on the text and while they were better at the semantic cues, they could not find the rhythm of the speech very well. Recent research combined these two modalities to grasp the benefits of both of them to synthesize more natural and semantically aligned gestures [11][6][4][53].

In this research, we are going to use both audio and transcript and will try to find their relation with the gestures in the latent space. Also, because of the idiosyncratic nature of gestures, we are going to interpret the gesture style using contrastive learning. In the next section, we discuss the previous attempts that have been made in this case.

2.4.2 Gesture style

Because each person performs the gestures differently many papers applied speaker identity in their gesture generation algorithm so their method can learn about different styles. However, because of the ambiguity of the relationship between the speaker and the gesture, different papers used different approaches to analyze that. Ao et al. [35] categorize these approaches into label-based and example-based groups.

In the label-based approaches, each sample is annotated with a label that specifies the style of that gesture. These labels can be as simple as a number [6], specifying the emotion [10], or describing a posture [54]. The problem with this approach is the limitation in the variety of labels and the difficulty of adding a new one.

On the other hand, example-based approaches can extract a style from a sample and mimic that to generate new samples with the same style. These samples can be a gesture sequence [32] or a video clip [3]. Although this approach is more flexible in the variety of styles, sometimes it is hard to judge whether the generated results are aligned with the example or not.

Zhi et al. [23] considered an id for each speaker that is converted to an embedding vector s which is learned through reparametrization. Also, they argued that because of the idiosyncratic nature of the gestures, they used the Kullback–Leibler divergence [55] to regularize the distribution. However, there are some downfalls with using KL in these situations like high sensitivity to discrepancies and mode collapse that can cause problems with unseen styles and learning from all styles respectively.

Ao et al. [35] proposed an approach that using CLIP guidance can extract the style from different modalities like text prompt, video, and gesture motions. They add the extracted style using adaptive instance normalization(AdaIN) [56] to their latent diffusion model. AdaIN simply tries to make the mean and variance of the generated results equal to the style sample.

In contrast to Ao et al. [35] was trying to find a relation between gesture and other modalities, we propose the idea of finding gestures within the gestures and in relation to the audio and context of the speech. So, in this project, using contrastive learning we try to make the style of the samples from one speaker close to each other while making them as far as possible to the style of the other speakers.

Also, in contrast to the label-based approaches where the style is assigned to the gestures, here we learn the style through the samples of the speakers.

2.4.3 Semantic motion synthesis

Nyatsanga et al. [11] argue that generating semantically aligned gestures depends on the input modalities. Audio input can be useful for extracting the beat cues and temporal alignment, while text transcripts are more useful for extracting semantics.

We can categorize the previous methods that tried to use text and find a relation between that modality and gesture, into two main categories: contrastive methods and GPT-based methods. [17] While the contrastive approaches come from a task-specific point of view, the GPT-based methods usually come from a task-agnostic point of view. Also, while the contrastive approach focuses on the similarity of the joint embeddings, the GPT-based approach focuses on the patterns and probabilities. Here we brought some examples from both categories.

Liu et al. [3] used contrastive learning to find a correlation between the high-level audio feature and transcript to benefit from the linguistic cues that come with transcripts. They considered the embeddings of the temporally aligned high-level audio features and transcript as positive and the high-level audio features from the other clips as negative. They also did not want to lose the low-level and mid-level audio features, so they added those features as negative similarities.

Followed by Liu et al. [3], Ao et al. [35] used Contrastive Learning Image Pre-training [51] to align the latent representation of transcript and gestures, so that gestures can have the underlying meaning of transcript. CLIP method lately showed great results related to coupling the unlabeled images and captions. So, they thought it could also work on gesture and transcript as well.

Jiang et al. [57] proposed that we can see motions as a language as it usually semantically couples with the text. They introduced MotionGPT which applies language modeling on both motion and text in a unified manner.

Later, Zhang et al. [34] tried to use LLMs and GPT models to find the semantics from the transcript text. Their model works mainly on understanding the concept using the LLMs and retrieving the proper semantic gesture from the pre-recorded dataset. So, it still has some limitations about the diversity of the results, especially about different languages and cultures that use different kinds of gestures.

The research is going on in both of these categories at the moment. However, in this research, we use the contrastive approach because of its computational simplicity in comparison to LLMs, its compatibility with diffusion models that have shown great results in the generative domain, and its support for multi-modalities.

2.4.4 Temporal consistency

As deep learning methods usually generate gesture motions frame by frame or in short sequences and merge them in the end, providing temporal consistency between these frames or sequences is necessary. Early models focused on recurrent neural networks solved this problem using the autoregressive manner of these types of networks and seed pose, while newer methods focused on diffusion models did different tricks. In this section, we present some of these ideas.

Yoon et al. [6] using a gated neural network architecture generated two-second motion clips using a feature vector from three modalities of audio, text, and speaker identity in addition to the seed pose. Seed pose is the four last frames from the previous clip that come with the other feature vectors to help the model with temporal consistency. Liu et al. [3] followed the same approach and tried to find the spatial-temporal dependency using a bidirectional GRU network.

When diffusion models showed great results in image generation researchers decided to use them in gesture generation as well. However, in contrast to recurrent neural networks, diffusions are not autoregressive which makes it harder for them to keep the temporal consistency in gesture generation.

Zhu et al [7] came up with two ideas to deal with temporal consistency in the diffusion model. First, they added a transformer to capture the long-term audio-gesture temporal dependency. Then, they added a new module called Diffusion Gesture Stabilizer that by gradually annealing down the noise discrepancy helped to remove the temporal inconsistency between synthesized sequences.

Inspired by Zhu et al. [7], Ji et al. [58] introduced C2G2 to enhance the temporal inconstancy of the previous diffusion approach. In contrast to Zhu et al. [7] which was trying to solve the temporal consistency only in the sampling part, C2G2 used cross-frame attention in their VQ-VAE to capture temporal cues in the latent space of the gesture samples.

On the other hand, Zhi et al. [23] used an MLP-based network as the denoiser of their diffusion model, and added the timestep embedding in addition to other modalities to the inputs. Each MLP block in this architecture consists of one fully connected layer for temporal alignment and one fully connected layer for spatial alignment.

Ao et al. [35] proposed an auto-regressive latent diffusion model. This model in addition to speech content and style prompts is guided by the previously generated motion.

Mughal et al. [37] came up with the idea of temporal latent representation. In this idea, they split a long sequence of motion into some chunks and encode them separately, but using only one decoder, they try to decode the concatenation of the encoded chunks into the origin motion sequence.

In this research, we follow the auto-regressive approach and use the embeddings of the last four frames of the previous sequence to guide our diffusion model. We train a linear layer during the training of our diffusion models to encode the last four frames of the previous sequence into a vector with a size of 512.

2.5 Materials

2.5.1 Contrastive learning approach

Contrastive learning is a machine-learning technique that is usually used for unsupervised or self-supervised learning. The basis of this technique is to learn representations by contrasting the positive and negative pairs. Positive pairs are the ones that are related to each other, and negative pairs are the non-related ones. For example, in our training, we expect the samples of the gestures of one speaker to share the same style and be different from the others. So, the samples of the same speaker are positive, and the samples from the other speakers are negative pairs.

In recent years, contrastive language image pre-training [51] became so popular as it showed great results for image captioning by learning from unlabeled image data. The CLIP model learns the similarities in images and related text and by contrasting them to the other images and texts. This success convinced the other researchers to use this technique in other domains.

We can see contrastive learning in co-speech gesture generation as well. Ao et al. [35], Deichler et al. [17], and Liu et al. [3] are some of these attempts. They tried to find similarities in the representation of audio and text, or text and gesture to find semantic cues from the text.

In this research, we use contrastive learning to find similarities between the audio, text, and gesture styles. During our data analysis, we realized the difference between the distribution of the embeddings of audio and text. So, in contrast to Deichler et al. [17] that used the concatenation of the representations of audio and text, we calculate the contrastive loss of these modalities separately each with a specific part of the gesture representation.

2.5.2 Dataset

In this section, two main approaches that have been used to collect gesticular data have been presented. After explaining each approach, some of the well-known ready datasets have been analyzed and reviewed.

Regarding data collection, there are two approaches. Extracting body poses and facial expressions can be done in a pseudo approach(Pose estimation [8]) from the in-the-wild videos [59][28][6] or in the motion capture lab by recording the position of body joints [60][61][12]. Although the first approach is cheaper and faster which makes it more suitable for creating large datasets, the second approach is much more accurate because of the several cameras capturing the subject from different angles and can be controlled in the laboratory environment [10].

It is also worth mentioning that early models used to work with datasets containing the 3D joint positions in Cartesian coordinates, while the latest models usually use the normalized direction vector of the body joints. [16]

We performed an investigation on the currently available datasets containing gesture motions, the way they have been collected, and what modalities they support. The results can be seen in Table 2.1.

In Table 2.1 we can see some of the best existing Speech-gesture datasets regarding the size of the data, number of participants, and diversity in modalities. As we can see BEAT [10] has a relatively large size, various number of speakers, and supports different modalities. Also, this dataset concludes high-quality fingers which is rare in the existing datasets.

On the other hand, TED [62] has a larger size and more speakers, but, it only concludes 2D positions of the body joints without fingers. PATS dataset [33] also is very large, but regarding the diversity of the speakers,

Name	Size	Nr of sp	Mot. format	Modalities	fg	Dialogue?
BEAT	76h	30	3d joint rot.	Ges, Audio, Text, Gest. properties, Emotion	yes	Both
TED Gesture	97h	1,776	3d joint dir.	Ges, Audio	no	Monolog
TED Expressive	97h	1,776	3d joint dir.	Ges, Audio	yes	Monolog
Talking With Hands GENEA Extension	20h	17	3d joint rot.	Ges, Audio, Text	yes	Dialogue
PATS	250h	25	2d coords.	Ges, Audio, Text	No	Monolog
Gesture-Speech Dataset	5h	2	3d joint rot.	Ges, Audio	yes	Monolog
Trinity Speech-Gesture I GENEA Extension	6h	1	3d joint rot.	Ges, Audio, Text	No	Monolog
SaGa	280m	1	3d joint rot.	Ges, Audio, Text, Ges. properties	No	Dialogue

Table 2.1: Speech-gesture datasets. "Nr of sp" stands for number of speakers, "Mot. format" stands for motion format, "fg" stands for supporting fingers, and "Dialogue?" specifies if conversations were held as dialogue, monologue, or both

it has fewer speakers in comparison to BEAT and TED which makes it hard to grasp different kinds of gestures that different people use for conveying the same meanings. Talking with hands [9], Gesture-speech [63], and Trinity [64] are also high-quality datasets that are relatively small.

In this research, we use the TED Expressive and BiGe datasets as diffusion models and contrastive learning is data-hungry and needs large datasets. Also, there are a decent amount of methods trained on the TED-related datasets, which makes it easier for benchmarking. However, because of the noise in these datasets, we faced some challenges.

2.5.2.1 Pose estimation

There are several researches to extract the position of the body joints from the in-the-wild videos. Early researchers aimed to do this in 2D using Openpose [65] like PATS [33], Speech-Gesture [59], and TED [62]. However, these

2D data were not able to completely guide the virtual 3D characters. So, researchers tried to extract 3D estimation of the body joints [28]. Because of the automatic nature of the process, the pose estimation approach is much cheaper and more scaleable than the motion capture approach [10], but sometimes not accurate and contains some noise as a result.

2.5.2.2 TED Gesture

The TED dataset [62] is one of the largest pose-estimated datasets. TED is a conference where people talk about their ideas. The videos of these talks and transcripts are available online to the public. So, Yoon et al. generated a dataset using OpenPose [65] from these videos. OpenPose is an algorithm that can extract 2D human poses from videos. The initial version was created from 1295 videos with an average length of 13 minutes. After resampling the videos to 15 fps, they selected about 12.9% of the shots where an informative body skeleton could be extracted from the shots using some algorithms. Also, they aligned the transcripts with the video timestamps using Gentle Library [66].

Finally, 52.7h data was gathered which contains the direction vector of 10 body joints, audio, and aligned English transcripts. Later, they added 471 additional videos to this dataset and using a 3D human pose estimator [67] converted 2D positions into 3D. Finally, The duration of the dataset was increased to 97h.

2.5.2.3 TED Expressive

While the TED Gesture dataset is relatively large and has diverse gesture styles from different people, the pose estimation algorithm that has been used to extract the body joints is not state-of-the-art anymore. So, Liu et al. [3] used ExPose [68] to extract 3D human body joints. Using this pose estimator they managed to extract 13 upper body joints and 30 finger joints which is more expressive than the TED Gesture skeletons. Furthermore, they converted these 43 3D coordinated into unit direction vectors which helps to minimize the effect of different body sizes. As the authors didn’t mention anything more about data mining, we can guess they used the same 1,766 videos that have been used to create the TED Gesture dataset.

2.5.2.4 BiGe

Hendric Voß and Stefan Kopp [22] also aimed to create a more diversified gesture dataset that supports full-body skeletons and fingers. So, in addition to the videos from the TED channel, they included videos from TEDx which concludes videos from other languages than English. In total, they collected 4327 videos with their transcripts with a total length of 1021 hours. Next, they modified the Yoon et al. [62] pipeline to extract 3D body joints, split them into clips, and select the shots of interest. For pose estimation they used AlphaPose [69], VideoPose [67], MediaPipe [70], and FastPose [71] to extract 59 3D full-body joints. In the end, from 2756, 54.360 shots were selected with an average length of 17 seconds to shape a dataset with a total duration of about 260h.

In comparison to TED Gesture and Expressive datasets, this dataset has more samples and uses better pose-estimation algorithms for extracting more body joints. However, the spectrogram of the audio is not provided by default in this dataset. Another difference in these datasets is that TED Gesture and Expressive annotated the words by their start and end time in seconds, while the BiGe dataset annotated them with the start and end frame number.

Unfortunately, due to the fire accident at their university, some parts of this dataset have been lost.

2.5.2.5 SaGa

SaGa [72] is another pose-estimated dataset that was created from 25 direction-giving dialogues in the lab. They used three synced cameras to capture the router, the follower, and both of them at the same time. The result contains 280 minutes of video material, 4961 gestures, and 39,435 words.

SaGa classified gestures into eight categories: Indexing, placing, shaping, drawing, posturing, sizing, counting, and hedging. Also, they considered the morphology of the handshape, wrist position, back of hand orientation, palm, and the movements of each of them.

They also provided the transcript of the speeches and categorized each clause into four categories: Landmark position description, landmark construction description, landmark property description, and naming a landmark. At the moment, SaGa is one of the best datasets regarding annotation, but it is in the German language.

2.5.2.6 Motion capture

Some other datasets are generated in the lab using motion capture equipment and players wearing special suits. Some of these datasets even used professional actors or directors to control the expressiveness of the gestures. Trinity [12], Talking with hands [9], and BEAT [10] are examples of motion-captured datasets. They are more accurate and have predefined themes and speech texts [10], but much smaller than the pose estimated datasets, which makes them not useful for data-hungry algorithms [8].

2.5.2.7 BEAT

BEAT is one of the largest motion-captured datasets at the moment. This dataset consists of body expressions and transcripts of 30 speakers, which were annotated in 8 different emotional categories. The total duration of this dataset is 76 hours which contains 2500 topic-segmented sequences. Actors used 4 different languages and talked in pairs which made it possible to use this dataset for training the speaker characters as well as the listeners [10].

2.5.3 Encoders

As encoders play a significant role in extracting the features from the input modalities, in this section a brief description of a couple of them has been presented.

2.5.3.1 Text encoders

2.5.3.2 BERT

In 2019, Devlin et al. [73] introduced BERT or bidirectional encoder representation from transformers which using masked language modeling (MLM) was able to make a deep bidirectional representation from an input text. Before that, language models were doing the pre-training in two different approaches. Radford et al. [46] did the pre-training in a unidirectional way from left to right on unlabeled data and fine-tuned the model for specific tasks. Peters et al. [74] did the pre-training separately in left to right and right to left, then used them as features for another model for a specific task.

However, in the BERT model, some parts of the input text are masked and the model learns to predict these masked words from the information on the left and right side of them. This bidirectional attention helps the model to grasp a better understanding of the context and generate better representations.

2.5.3.3 Audio encoders

2.5.3.4 Wav2Vec2

Wav2Vec2 [75] was invented by a team from Facebook for speech recognition purposes. They realized that learning the generalized representation of speeches from some un-labeled speech audio and fine-tuning that with limited transcribed audio can increase the accuracy in speech recognition tasks in comparison to semi-supervised methods.

Regarding the architecture, a couple of Temporal Convolutional layers have been used to encode a sequence of raw audio into a latent space $\mathcal{Z}$. Then, Transformers find the contextual relations of the encoded audio from the entire sequence to encode them into a contextual latent representation $\mathcal{C}$. Also, during self-supervised learning, the encoded features $\mathcal{Z}$ are discretized into a finite codebook using product quantization. They used a contrastive loss to help find the true quantized speech representation given a contextual representation c, and a diversity loss to help using all the codebook.

2.5.3.5 HuBERT

Inspired by BERT [73] and Wav2Vec2 [75], Hsu et al. [76] made an audio representation using mask prediction technique. In contrast to text input which is discrete, audio is continuous and needs to be classified or contrasted against other negative samples. Also, in contrast to computer vision classification tasks where each image is a unit by itself, input audio can contain different sounds at the same time and can't be labeled as an instance.

So, to overcome these limitations related to audio, in the HuBERT architecture, the audio is clustered using K-Means and an index of a codebook will be assigned to each cluster. Then, some parts of the audio are masked and the model learns to predict the indices of the masked part. The clustering iteratively gets better by the latent representation learned from the mask prediction and clustering and predicting the mask part improve each other simultaneously.

The HuBERT encoder showed better or equal results to Wav2Vec2 in the speech recognition tasks and was widely used in the co-speech gesture generation. [76]

2.5.3.6 Data encoder

2.5.3.7 Data2Vec

In contrast to the prior works that tried to generate a latent representation from a single modality, Baevski et al. [77] invented a general approach to conclude three modalities of text, audio, and image. Although they have used different encoders and masking strategies for different modalities, the learning regime for latent representation is the same for all three modalities.

In this architecture, after building contextual representations of the full input as the target (teacher), a masked version of the input will be encoded (student). The goal of the training is to predict the representation of the whole input based on the encoded representation of the masked input while the weights of the teacher are updated by the weights of the student.

In comparison to the prior models, Data2Vec learns more about the context of the modalities, as it predicts the representation of the whole input, while the BERT or Wav2Vec2 only predicts the masked tokens during the pre-training.

We used this encoder because of its unified approach to different modalities that could be helpful in our model that works with similarities. Also, this encoder showed good results regarding the contextual feature extraction that is helpful for generating semantic gestures.

2.6 Evaluation

2.6.1 Objective Evaluation

Although different metrics have been developed to evaluate generated gestures, still evaluating human likeness and semantic relatedness of the results is not accurately possible [10]. One of the reasons is the idiosyncratic and non-periodic nature of gestures that leads to different gestures for the same speech from different people or even the same person in different environments or different inner conditions like emotions [11]. However, researchers trying to compare the generated body joints to the ground truth.

Liu et al. [10] introduced the Semantic Relevance Gesture Recall that instead of using L1 or L2, it calculates the probability of the position of the correct body joints. In their subjective experiment, they asked the participants to rate the generated and ground truth gesture animation based on diversity and attractiveness. In this experiment, the results from SRGR were closer to the collected results than the results from L1 distance.

$$D_{SRGR} = \lambda \Sigma \frac{1}{T \times J} \sum_{t=1}^{T} \sum_{j=1}^{J} 1[\|p_t^j - \widehat{p}_t^j\|_2 < \delta] \tag{2.1}$$

Where 1 is the indicator function and T, J is the set of frames and number of joints. Also, δ is the threshold. However, different people use different gestures for the same subject, or even the same person might perform different gestures for the same speech in different situations. So, evaluating the semantic alignment based on the location of the joints might not be a good solution.

Another metric for finding how aligned is the generated gestures with the speech is BeatAlign. [78] This metric is a Chamfer distance between the audio and gesture beats.

$$BC = \frac{1}{n}\sum_{i=1}^{n} exp(-\frac{\forall {t_j}^x \in B^x \|{t_i}^x - {t_j}^y\|^2}{2\sigma^2})$$

Where $B^x = \{{t_i}^x\}$ are kinematic beats, $B^y = \{{t_j}^y\}$ are audio beats, and σ is a parameter to normalize sequences based on fps.

Heusel et al. introduced Fréchet Inception Distance (FID) which has been generally used for evaluation of the general motions [79][80]. Yoon et al. [6] introduced Freshet Gesture Distance which is an adoption of FID for gesture motions. In this metric, the Gaussian mean and covariance of the latent features of the generated gestures are calculated.

$$FGD(X, \hat{X}) = \|\mu_r - \mu_g\|^2 + \mathrm{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}) \tag{2.2}$$

where μ_r and Σ_r are the first and second moments of the latent feature distribution Z_r of real human gestures X, and μ_g and σ_g are the first and second moments of the latent feature distribution Z_g of generated gestures X [6].

We also calculate the MAE of the ground truth and generated gestures in the latent space.

$$MAE = \frac{1}{M}\sum_{j=1}^{M} |r_j - g_j| \tag{2.3}$$

Where M is the number the length of latent space, and r_j and g_j are the j^{th} elements of the real and generated latent vectors.

During this research, we realized the high impact of hand joints on this metric while the body joints might be more important in the naturalness and semantic grounding of the gestures. Because of the high number of joints in the hands, the latent gesture vectors are also more influenced by these joints. This issue will be more critical when it comes to training on the TED expressive dataset where because of the pose estimation algorithm that has been used for creating this dataset, there is so much noise in the hand joints.

Finally, we have a diversity metric that measures how each model can generate diversified gestures regarding the position of the body joints. To compute this we find the average Euclidean distance between the features of all possible motion sequences. In this project, we considered 500 of the generated samples $F1 = [f_1, f_2, ..., f_{500}]$ and shuffled it to have $F2 = [f'_1, f'_2, ..., f'_{500}]$.

$$\text{diversity} = \frac{1}{500} \sum_{i=1}^{500} \sum_{j=1}^{M} \left| f_{ij} - f'_{ij} \right| \tag{2.4}$$

2.6.2 Subjective Evaluation

Although some metrics have been developed to evaluate the generated gestures, they can not examine human perception which is critical in this subject. So, subjective evaluation usually comes along with objective evaluation in this field [10][11]. One of the subjective evaluation approaches is matched vs mismatched inspired by the Turing test. In this test, one gesture is generated for a target speech and one gesture is generated for another one. Then, participants are asked to choose which gesture is matched with the target speech. In another scenario, the participants are asked to choose which gesture is generated and which one has been recorded [81].

3. Research Questions

3.1 Research questions

After investigations on previous works that have been done to synthesize co-speech gestures, we realized that still there is a gap in synthesizing semantically aligned gestures. So, the goal of this research is to explore:

- How semantically aligned gestures can be generated?
- Whether using contrastive learning for generating co-speech gestures can help to enhance semantic alignment in comparison to the state-of-the-art methods?
- How semantic alignment of the gestures can be evaluated? What are the appropriate subjective and objective evaluation metrics?
- Are existing gesture datasets sufficient for generating semantically grounded gestures? How that aspect can be analyzed?

Therefore the main goal of the project is to generate realistic, temporally and semantically aligned gestures from the speech. Previously we observed that using CLIP-based methods can improve the quality of the generated motions from text [57], so we decided to use contrastive learning for gesture synthesis. Also, we want to know how we can evaluate the semantic alignment of the gestures. In the end, we examine whether we have enough data to generate semantically aligned gestures that can be interpreted by different people from different nations.

4. Data Analysis

4.1 TED Expressive dataset

We ran an analysis on the TED Expressive dataset to gain an overview of the data distribution in this dataset. As recent papers showed, modeling the hand and body separately can enhance the results, so we analyzed them solely and combined them as a whole gesture. Also, the embeddings of audio and transcripts of samples encoded by the data2vec pre-trained model were added to the dataset to see whether we could find a correlation between these modalities and gestures, or not.

In the figure 4.1, we can see the distribution of normalized direction vectors of hand and body joints separately. This figure shows the difference in the distribution of movement of hand and body joints as the shape of the distribution, scale, and the maximum and minimum in each direction are different. Also, in contrast to body distribution which is smooth and evenly distributed, the distribution of the hand shows some anomalies and gaps in some spaces which is an indicator of the noise in this part. The pose estimation algorithm used for this dataset does not show promising results in the hand joints, and it can be observed both in the distribution of data and the visualized videos from the dataset. In addition, we compared the sequences of body and hand joint direction vectors consisting of 34 frames with each other. In the figure 4.2, we can see that when the $(34, 126)$ and $(34, 33)$ vectors of body and hand sequences respectively are mapped into 2D space using PCA, their distributions are different. These differences in the distribution of the normalized direction vector of hand and body joints convinced us to model them separately and combine the generated results.

In the next step, we analyzed the distribution of audio and transcript embeddings and compared them with the distribution of direction vectors

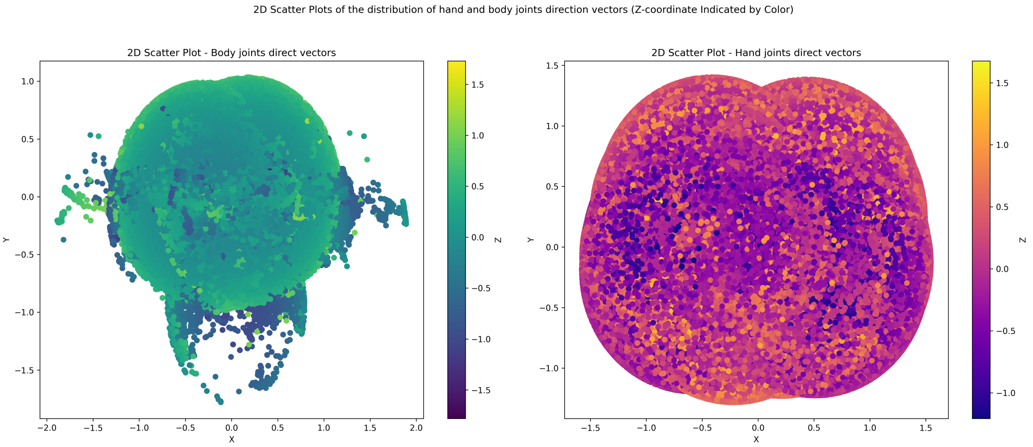


Figure 4.1: Scatter plot of the distribution of normalized direction vectors of hand and body joints in the TED Expressive dataset

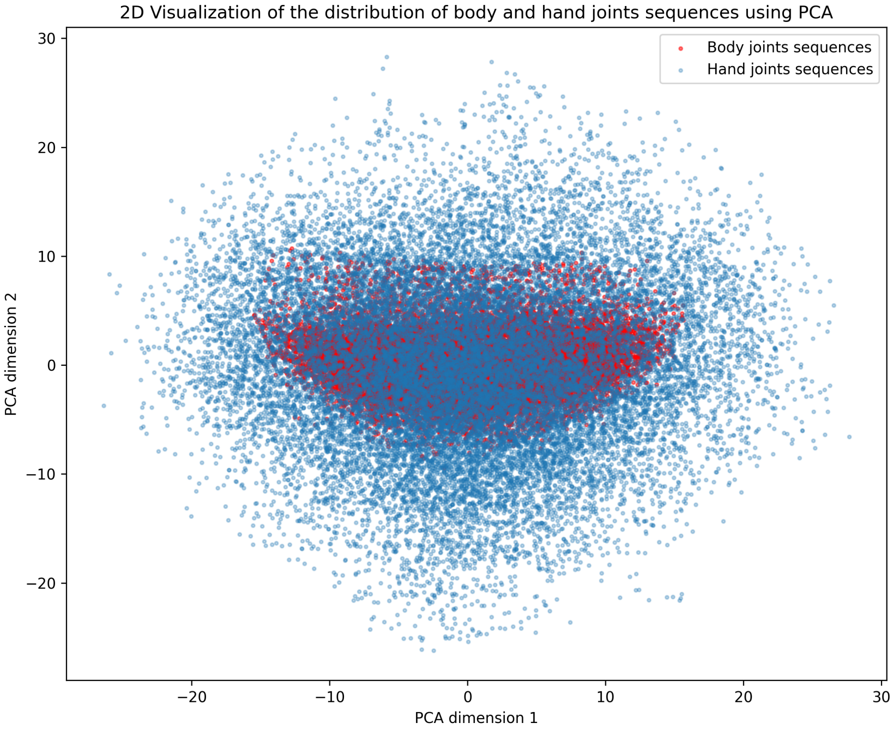


Figure 4.2: 2D visualization of the distribution of body and hand joints direction vectors in 34 frame sequences

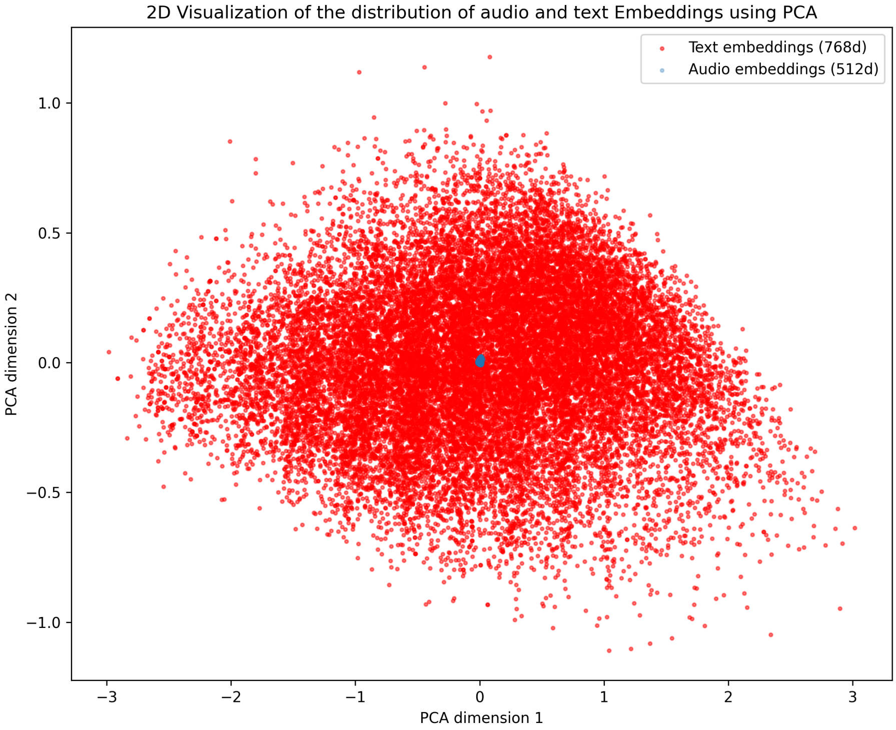


Figure 4.3: Visualization of the distribution of audio and transcript embeddings using PCA

of hand and body joint sequences. In the figure 4.3, we can see the distribution of audio and transcript embeddings has a huge distance, and while audio embeddings are concentrated in the middle, transcript embeddings spread around.

By observing this distance between the embeddings of these two modalities, we realized that using the concatenation of audio and transcript embeddings in contrastive learning like [17] might not be a good idea as the embedding of gestures cannot easily get closer to two completely different embeddings.

We also compared the audio and transcript embeddings with the hand and body direction vector sequences, but we did not observe many similarities that show the importance of contrastive learning in this subject. Pairwise visualization of the comparison of these modalities can be seen in the figures 4.4, 4.5, 4.6, and 4.7.

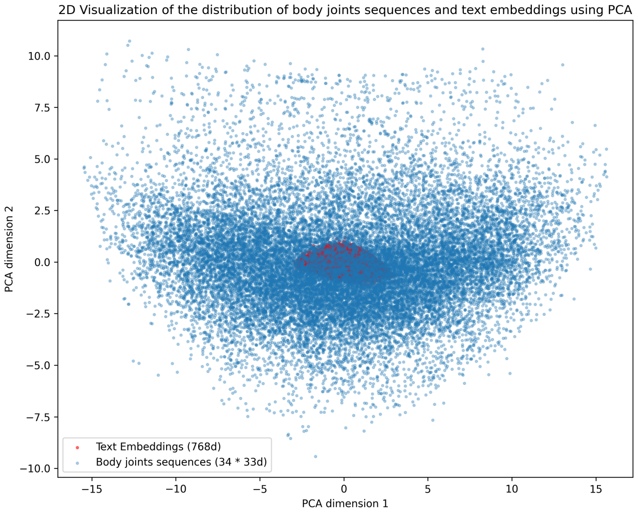


Figure 4.4: Visualization of the distribution of text embeddings and body joint sequences using PCA

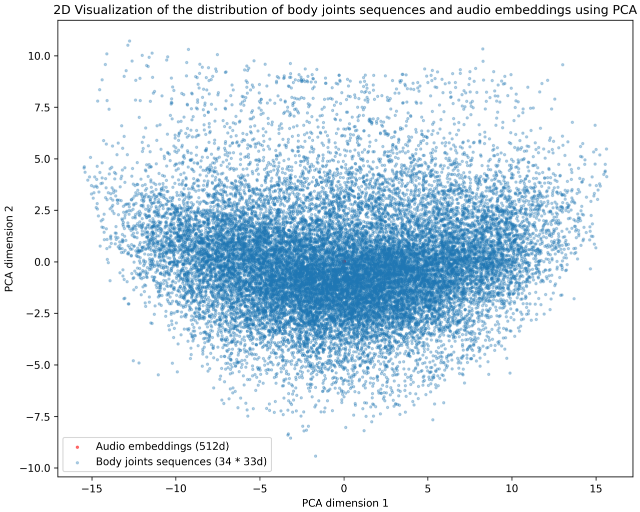


Figure 4.5: Visualization of the distribution of audio embeddings and body joint sequences using PCA

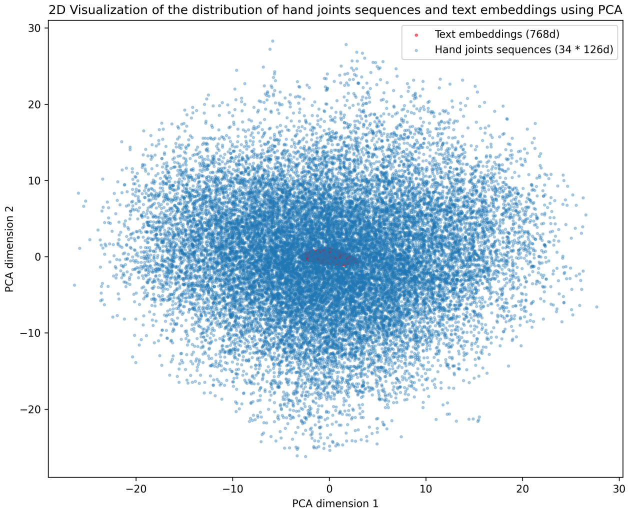


Figure 4.6: Visualization of the distribution of text embeddings and hand joint sequences using PCA

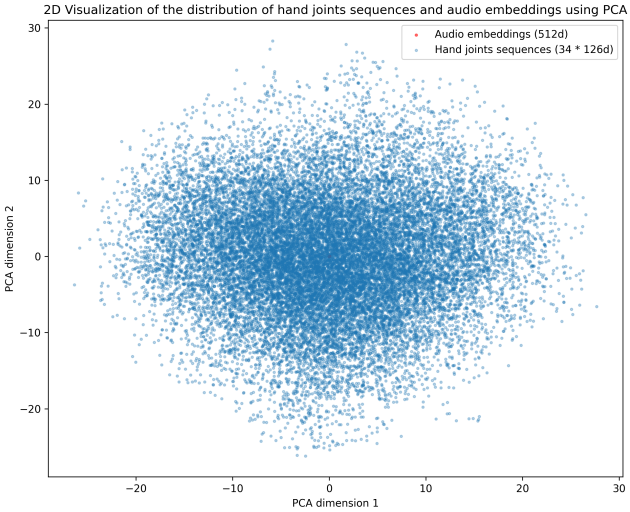


Figure 4.7: Visualization of the distribution of audio embeddings and hand joint sequences using PCA

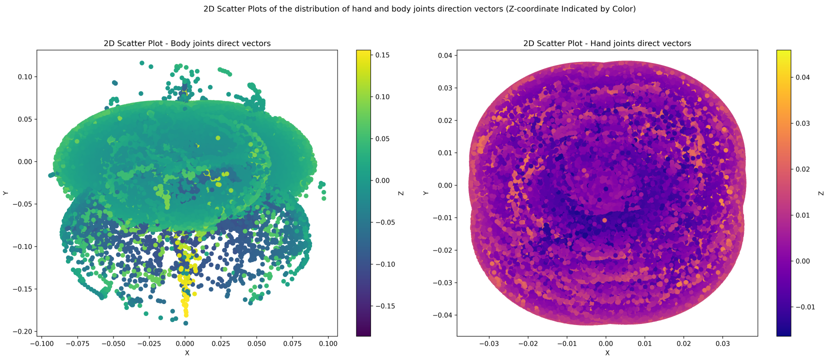


Figure 4.8: Scatter plot of the distribution of normalized direction vectors of hand and body joints in the BiGe dataset

4.2 BiGe dataset

We ran the same analysis on the BiGe dataset and almost got the same results which can be because of the same approach (pose estimation) that has been acquired to create these datasets and the same origin of videos. The distributions of the direct vector of hand and body joints are neither the same nor similar to the distribution of audio and transcript embeddings.

However, there are some differences. The first one is the order of joints in each sample sequence. While the body joints in the BiGe dataset come first and hand joints are located on the following indexes, the location of body joints in the TED Expressive dataset is not this simple and needs to be masked with the index numbers.

The other difference is the scale of dimensions in the direction vector of body and hand joints in these two datasets. While the x, y, and z vary mostly between -1.5 and 1.5 in the TED Expressive dataset, in the BiGe dataset, dimensions of hand joints vary between -0.04 and 0.04, and dimension of body joints between -0.1 and 0.1. This difference in the dimensions of body and hand joints can emphasize on the importance of modeling and training them separately.

5. Method

5.1 Problem Formulation

Given an audio speech, the transcripts, and the speaker identity, we want to generate realistic, diverse, and semantically aligned gestures. The training audio, timestamp-aligned transcripts with paddings, and gestures are cut into clips with a sliding window. For each N-frame clip, all modalities are encoded to features. Audio $A = [a_1, ..., a_n]$, gesture G = $[g_1, ..., g_n]$, transcript $T = [t_1, ..., t_n]$.

$g_i \in \mathbb{R}^{3J}$ where J is the total joint number. The training objective is to generate a motion clip(M) by giving audio(A) and related transcripts(T). At the inference stage, generated motion clips from audio and transcripts are smoothly connected to create realistic, temporally, and semantically aligned gestures.

5.2 Data pre-processing

We followed the data pre-processing method of previous researchers like Zhu et al. [7] and Voß et al. [82] to create some sequence samples from TED Expressive and BiGe datasets. In this process, each sample in the dataset that contains the joint direction vectors, audio, transcript, and some information about the video will be split with a stride of 10 into some sequences with 34 frames(15 fps).

We cache these sequences that contain the normalized joint direction, audio of that timestamp, and spoken words within that sequence, in addition to the embeddings of audio and transcripts for speeding up the process.

For using audio and transcript of the speeches we need to convert them into latent space. We used pre-trained data2vec [77] encoders and tokenizers for this purpose. Encoding the audio is pretty straightforward, and using $embeddings$ layer of pre-trained "$facebook/data2vec - audio - base - 960h$" checkpoint of the $Data2VecAudioForXVector$ model we can get a vector with the size of 512 from the audio.

The process of converting the transcripts into the latent space is more complicated. One of the reasons is that the number of words in each sample is varied based on the speed of the speech and the pauses in between. So, after finding the related words in that sample, they are concatenated so that during the encoding we can grasp the underlying information that comes within the relation of words that come together. Then, using the $AutoTokenizer$ they are converted to IDs and after adding some padding to unify the length of them, they can be encoded by the data2vec text model. We take the output of the last hidden layer of the pre-trained "$facebook/data2vec - text - base$" checkpoint of the $Data2VecTextModel$. The length of this output is 768 which is converted into 512 by a linear layer we train during the training of the VQ-VAE model.

5.3 Model

We found that models that tried to use text as input gained higher scores regarding the semantics compared to the models that only used audio for gesture generation [11]. So, in contrast to Zhu et al. [7], which used initial pose, timestamp, and audio as the inputs, we use the transcript of the speeches as well to generate more semantically aligned gestures.

Also, to make a connection between these different modalities, like the previous researches [35] [17] [23], we used a contrastive loss. However, with some considerations that we believe can improve the impact of this technique. These considerations are: specifying a specific place in the latent gesture vector to be aligned with the audio, transcript, and style of the speaker. And, using contrastive loss separately for each modality and their related part in the latent gesture vector.

Furthermore, gestures are idiosyncratic, and different people might make

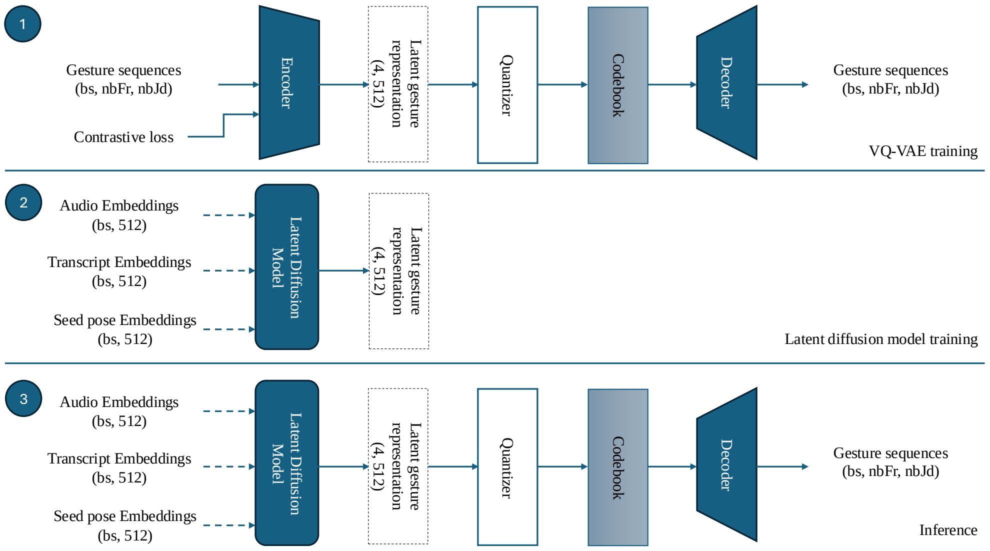

1
Gesture sequences (bs, nbFr, nbJd)
Contrastive loss
Encoder
Latent gesture representation (4, 512)
Quantizer
Codebook
Decoder
Gesture sequences (bs, nbFr, nbJd)
VQ-VAE training
2
Audio Embeddings (bs, 512)
Transcript Embeddings (bs, 512)
Seed pose Embeddings (bs, 512)
Latent Diffusion Model
Latent gesture representation (4, 512)
Latent diffusion model training
3
Audio Embeddings (bs, 512)
Transcript Embeddings (bs, 512)
Seed pose Embeddings (bs, 512)
Latent Diffusion Model
Latent gesture representation (4, 512)
Quantizer
Codebook
Decoder
Gesture sequences (bs, nbFr, nbJd)
Inference

Figure 5.1: An overview of our methodology. Step 1 is about training a VQ-VAE model that is guided by a contrastive loss. Step 2 is about training a latent diffusion model conditioned on audio, transcript, and seed pose embeddings that generate latent gesture representations. Step 3 is about inferencing. Using the trained diffusion model, codebook, and decoder, we can generate gesture sequences from the speech. bs, $nbFr$, and $nbJd$ are batch size, number of frames, and number of joint direction vectors respectively.

different gestures for the same speech. Or even one person can perform different gestures for the same speech in different situations [11]. In this research, we considered one speaker almost keeping the same style of gesture for the short duration of their speech. So, for the emphasis on the idiosyncratic nature of the gestures, different IDs for different speakers were considered. Then, using contrastive loss we tried to make the latent gesture vector of the same speaker closer to each other and make the latent gesture vector of different speakers as far as possible.

Here, we want to know whether using contrastive learning separately on audio, transcript, and gesture style can improve the generation results regarding the naturalness and semantics or not.

5.3.1 Model Architecture

Our model consists of two parts. The first part makes a codebook of latent representation of the gesture motions using the VQ-VAE [44] algorithm that is guided by a contrastive loss. In the second part, we train a latent diffusion model conditioned on the embeddings of audio, transcripts, and seed pose. Finally, using the codebook we learned in the first step, we will decode the generated latent representations to the gesture sequences. The overview of this method can be seen in figure 5.1.

Chen et al. [83] emphasized the importance of using motion latent space instead of raw motion data for generating general motions because of the computational complexity of working on the raw data and noise in the datasets.

So, in this step, we try to train a VQ-VAE model to represent raw gestures as tokens in a latent space. For this means, we adopted the architecture of T2M-GPT [80] one of the state-of-the-art methods in motion generation, and modified it to our needs.

This VQ-VAE model consists of an encoder, quantizer, and decoder where the encoder $\mathcal{E}$ gets a sequence of gesture motions $g = g_0, ..., g_N$ where N is the length of the sequence, and turns them into a latent vector $L_g = l_0, ..., l_M$ where M is the length of the latent vector. At the same time, the decoder $\mathcal{D}$ learns to generate gestures g from the quantized version of the latent vector L_g. These encoder and decoder consist of some 1D convolutional layers, ReLu activations, and residual connections. The number of layers L can be changed depending on the temporal down-sampling we want. The diagram of this architecture can be seen in the figure 5.3.

We added a normalization layer at the end of the encoder and normalized the latent gesture vectors to stabilize the training and help the diffusion model to predict them in the next step. An evaluation on the performance of this decision is presented in the section 6.1.3.

To train this model, we use three loss functions to monitor reconstruction $\mathcal{L}_r$, commitment $\mathcal{L}_c$, and contrastive loss $\mathcal{L}_{contrastive}$. The recreation loss checks if the output of the decoder is close to the input of the encoder. The commitment loss encourages the encoder to stay close to the chosen codebook index. And, the contrastive loss tries to align the gesture embeddings with the audio and transcript embeddings as well as the style of the gestures of the same speaker.

$$\mathcal{L}_{\mathcal{V}} = \mathcal{L}_r + \alpha \mathcal{L}_c + \beta \mathcal{L}_{contrastive} \tag{5.1}$$

Where α and β control the weight of commitment and contrastive loss. We specified α as the 0.1 of the best $\mathcal{L}_r$ we got without that, and β as 0.5 of the best $\mathcal{L}_r$ we got without the $\mathcal{L}_{contrastive}$. Also, for the $\mathcal{L}_r$, we tried $L2$ and $L1_smooth$ and realized $L1_smooth$ works better in this training.

$$\mathcal{L}_r = ||x - \mathcal{D}(e)]||_2{}^2 \tag{5.2}$$

$$\mathcal{L}_c = ||sg[e] - \mathcal{E}(x)||_2{}^2 \tag{5.3}$$

Where sg is a stop-gradient operation that prevents gradient from being applied to its arguments, and e is the selected codebook.

$$\mathcal{L}_{contrastive} = \mathcal{L}1_{contrastive} + \mathcal{L}2_{contrastive} + (\mathcal{L}3_{contrastive} + \mathcal{L}4_{contrastive})/2 \tag{5.4}$$

Where $\mathcal{L}1_{contrastive}$ calculates the contrastive loss between the transcript embeddings and first latent gesture vector, $\mathcal{L}2_{contrastive}$ calculates the contrastive loss between the audio embeddings and second latent gesture vector, and $\mathcal{L}3_{contrastive}$ and $\mathcal{L}4_{contrastive}$ calculate the contrastive loss between the third and fourth gesture latent vectors of different samples.

For this training, we set the number of residual blocks L to three, and the size of the embeddings to 512 which leads to four vectors with a length of 512. There is a trade-off in the number of residual blocks as when we increase them, we will have latent gesture vectors with lower dimensions that can be generated easier by our diffusion model, but at the same time, more information will be lost as we are decreasing the size of the latent space. So, we tried two, three, and four as the number of residual blocks that led to eight, four, and two gesture latent vectors respectively, and realized two latent vectors with the size of 512 might not contain enough information that we need. So, during the recreation using the decoder, there were some dissimilarities with the ground truth. On the other hand, eight latent gesture vectors are so much to predict by our diffusion model.

So, we chose three residual blocks to have four gesture latent vectors after encoding. One of these vectors was considered to contain the audio-related cues, the other one to contain the transcripts cues, and the last two for style-related cues. For this means, we used the contraSim [52] algorithm and the data2vec [77] audio and text encoders.

During our data analysis, we realized that the audio embeddings and transcript embeddings for the same samples from the data2vec encoders are not close to each other. So, in contrast to Deichler et al. [17] that used the concatenation of these embeddings in their contrastive algorithm, we used these two embeddings separately with different gesture latent vectors.

Using the contraSim [52] algorithm, we tried to make the first latent gesture vectors of the samples similar to transcribe embeddings and the second latent gesture vector close to audio embeddings. Also, we tried to make the third gesture latent vector of the samples of the same speaker close to each other while as far as possible from the samples of the other speakers. So, that we can learn the gesture style of speakers in that latent gesture vector. We did the same thing as the third gesture latent vector on the fourth latent gesture vector.

The benefit of this way of approaching gesture style is that this style is learned instead of forced into the model by an external modality, and as far as we know, it is novel in this area of research. The limitation of this approach is its dependency on the high number of samples. So, it would not be beneficial to train on the datasets with a limited number of speakers.

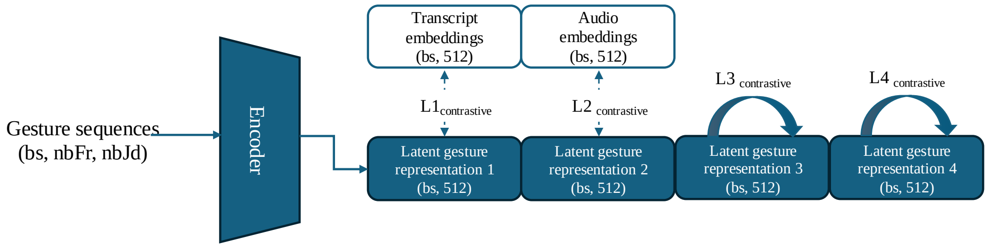

Transcript embeddings (bs, 512)
Audio embeddings (bs, 512)
L3 contrastive
L4 contrastive
L1contrastive
L2 contrastive
Gesture sequences (bs, nbFr, nbJd)
Encoder
Latent gesture representation 1 (bs, 512)
Latent gesture representation 2 (bs, 512)
Latent gesture representation 3 (bs, 512)
Latent gesture representation 4 (bs, 512)

Figure 5.2: The encoder of the VQ-VAE model converts the gesture sequences into four vectors with the size of 512. We used the contrastive loss to align the first two with the transcript and audio embeddings. The other vectors are aligned with the latent vectors of the other samples from the same speaker to depict the style of the gestures of that speaker. *bs*, *nbFr*, and *nbJd* are batch size, number of frames, and number of joint direction vectors respectively.

$$\mathcal{L} = \sum_{i \in I} \frac{-1}{|P(i)|} \log \frac{\sum_{p \in P(i)} \exp(z_i \cdot z_p / \tau)}{\sum_{n \in N(i)} \exp(z_i \cdot z_n / \tau)} \tag{5.5}$$

The formula 5.5 is our contrastive loss formula inspired by the contraSim paper [52] where $P(i)$ and $N(i)$ are the positive and negative samples, and z_p and z_n the normalized embeddings of positive and negative samples respectively. τ is the temperature that for the audio and transcript was considered 0.01 and for the style as one. Lower values of τ increase the magnitude of these scaled dot products, while higher values decrease them.

As studies showed before, using EMA [44] for the embedding loss and codebook reset techniques can improve the training of VQ-VAE models [80]. Regarding the calculation of EMA, we consider ${n_i}^t$ as the number of vectors in $\mathcal{E}(x)$, e_i as the codebook index, and λ as the decay factor between 0 and 1. So, we have:

$$N_i{}^t := N_i{}^{(t-1)} * \lambda + n_i{}^t(1 - \lambda) \tag{5.6}$$

$$m_i{}^(t) := m_i{}^{(t-1)} * \lambda + \Sigma_j^{n_i{}^t} \mathcal{E}(x)_{i,j}{}^t (1 - \lambda) \tag{5.7}$$

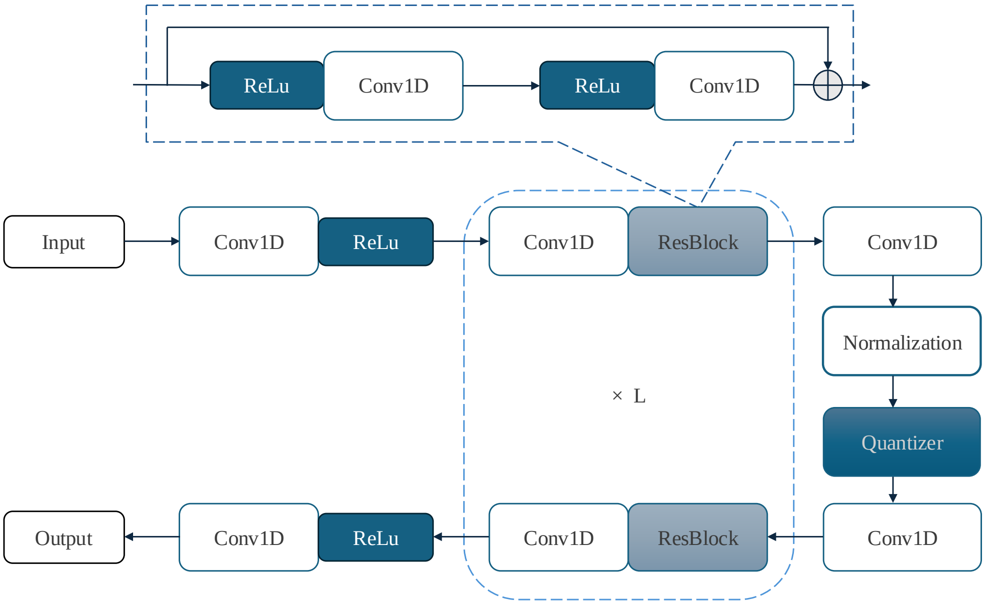


Figure 5.3: The architecture of the VQ-VAE model. The L shows the number of repetitions of convolutional 1D and residual block in that place. What is inside the residual blocks can be seen on top of the figure.

$$e_i{}^t := \frac{m_i{}^t}{N_i{}^t} \tag{5.8}$$

Codebook reset refers to resetting inactive codebook indices to make them active and use the maximum capacity of the codebook. For this means, followed by T2M-GPT [80], we update the less frequently used codebook indices with a random value:

$$\mathbf{C}_i = \mathbf{u}_i \cdot \mathbf{C}_i + (1 - \mathbf{u}_i) \cdot \mathbf{R}_i \tag{5.9}$$

Where C_i is the i-th code in the codebook, u_i is the usage indicator, and R_i is a new random vector.

After training the VQ-VAE model, we need a latent diffusion model [84][35] that is conditioned on audio and transcribe embeddings, in addition to seed pose to generate latent gesture vectors. Then, using the quantizer and decoder of the VQ-VAE model we can convert the generated latent gesture vectors into gesture direction vectors.

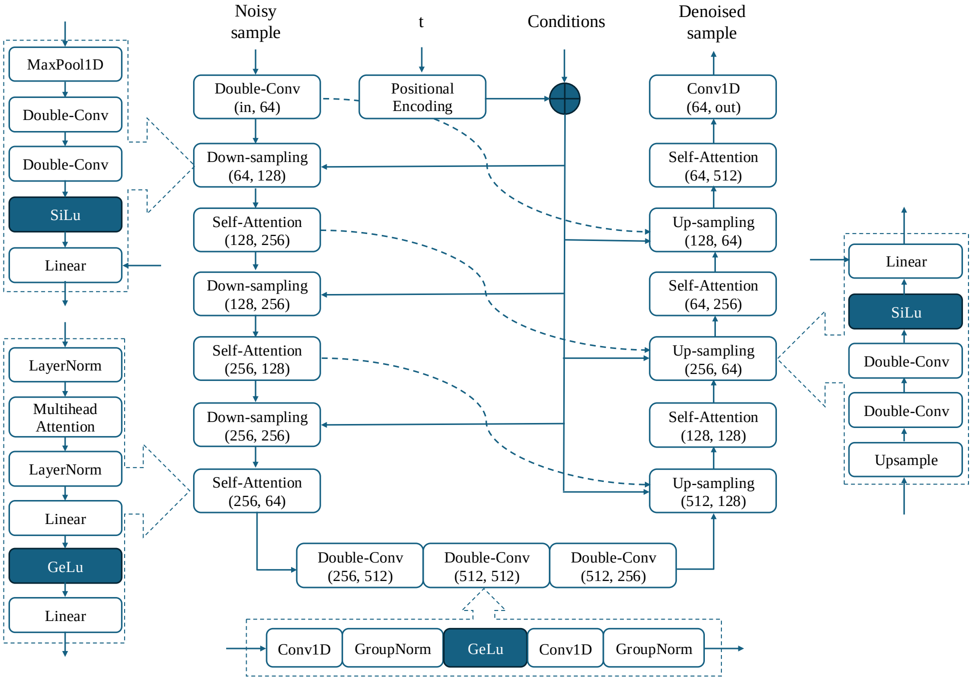


Figure 5.4: The structure of the denoiser of the latent diffusion model. It consists of some Down-sampling, Up-sampling, and Self-attention blocks that were presented on the sides. The timestamp is encoded using a positional encoder and is added to the conditions. Then, the summation is added to the Linear layer of the Down-sampling and Up-sampling blocks.

5.3.2 Latent Diffusion Model

Assuming $q(x)$ is an unknown density distribution of x, to construct a diffusion model upon x, in the first phase or forward phase, we need a Markov-chain $q(x_n|x_0)$ for $n \in [1, ..., N]$ that constantly adds Gaussian noise to x_0 so that $q(x_N, x_0)$ has an approximately standard normal distribution. In the second phase, the model is trained to generate observations based on the noise. This process is called denoising or reversing which starts from x_n, $x \in [1, ..., N]$, and at each step noise is gradually removed until step 1.

Assuming at each step zero-mean Gaussian noise is added to the previous step:

$$q(x_n|x_{n-1}) = \mathcal{N}(x_n; \sqrt{1-\alpha_n}x_{n-1}, \alpha_n I) \tag{5.10}$$

Where $\mathcal{N}$ is a multivariate Gaussian density function at step x_n, $n \in 1, ..., N$,

and α is a constant variable for enhancing the reverse phase.
When the noise added in step n is relatively small, we can assume the reverse distribution is also Gaussian:

$$p(x_N) = \mathcal{N}(x_N; 0, I) \tag{5.11}$$

$$p(x_{n-1}|x_n) = \mathcal{N}(x_{n-1}; \mu(x_n, n), \Sigma(x_n, n)) \tag{5.12}$$

Where $\mathcal{N}$ is a multivariate Gaussian density function at step x_{n-1}, μ is the mean, and Σ is a scaled identity matrix. Then, if we assume $\beta = 1 - \alpha$, and $\hat{\beta} = \Pi_{n=0}^{N}\beta_n$, the noisy gesture x_n at step n can be written as:

$$q(x_n|x_0) = \mathcal{N}(x_n; \sqrt{\hat{\beta_n}}x_0, (1 - \hat{\beta_n})I) \tag{5.13}$$

In sampling, we can learn the mean(μ) by:

$$\mu_\theta(x_n, n) = \frac{1}{\sqrt{\beta_n}}(x_n - \frac{\alpha_t}{\sqrt{1 - \hat{\beta_n}}}\epsilon_\theta(x_n, n)) \tag{5.14}$$

Where ϵ is the real noise and $\epsilon(x_n, n)$ is the predicted noise.

Our latent diffusion model works with a U-Net [85] that consists of some down-sampling, up-sampling, and self-attention [45] blocks. The down-sampling part is like a convolution network consisting of some convolution layers, max poolings, and activation layers trying to decrease the spatial information and increase the number of extracted features. The up-sampling is the opposite and using up-sampling layers tries to create the output using the extracted features. Details of this model can be seen in figure 5.4.

Furthermore, we used an attention mechanism to extract the temporal relations that come with all of our sequential input modalities. In our self-attention block, we used a multi-head attention module to focus on different parts of the sequence at the same time.

$$\begin{aligned}\mathrm{MultiHead}(\mathbf{Q},\mathbf{K},\mathbf{V}) &= [\mathrm{head}_1,\ldots,\mathrm{head}_h]\mathbf{W}_0\\ \text{where } \mathrm{head}_i &= \mathrm{Attention}(\mathbf{QW}_i^Q,\mathbf{KW}_i^K,\mathbf{VW}_i^V)\end{aligned}$$

To guide this diffusion model, we use a condition vector shaped by concatenation of audio and transcript embeddings and another vector with the length of 512 that is trained simultaneously to encode the last 4 frames of the previous sample as a seed pose. We use seed pose to improve the temporal consistency between the generated gestures.

The condition vector is added to the positional encoded timestamp and is fed to our U-Net network at the last layer of Down-sampling and Up-sampling blocks. Our objective here is that our diffusion model would be able to generate those latent gesture vectors that became similar to audio and transcript embeddings, using the guidance of embedding of these modalities.

Inspired by Mughal et al. [37], at each iteration, with a probability of 0.1, randomly we set one of the condition vectors equal to zero so that the model can learn from all of the conditions.

The loss function of our diffusion model is the distance of the predicted latent vector to the latent gesture vector coming from the encoder of our VQ-VAE model. We used l1 smooth for calculating this distance. Also, for the validation, using the quantizer and decoder of the VQ-VAE model we convert the predicted latent vectors to the joint direction vectors and compare them to the ground truth.

6. Results

6.1 Model selection

6.1.1 Embedding alignment

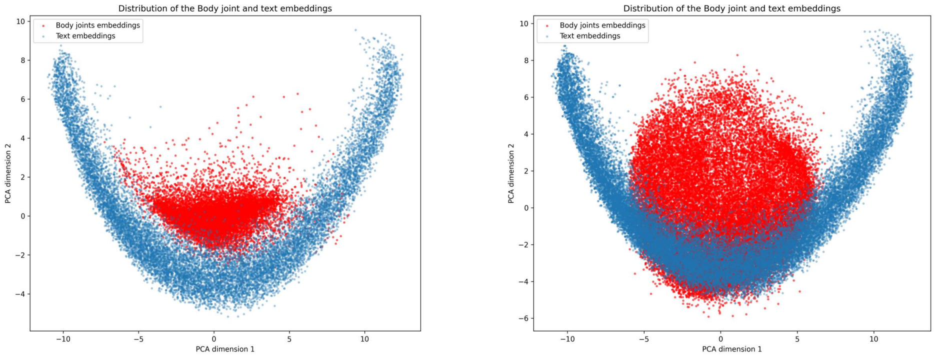
During the data analysis, we realized the difference between the distribution of the direction vector of the hand joints and other body parts. So, we decided to separate their training and train one VQ-VAE and latent diffusion model for each of them. Later we can combine the results and create the whole generated upper-body. This decision increased the training time to almost double, but the FGD of reconstruction with our VQ-VAE models was decreased to almost half.

Also, we monitored the distribution of the first and second latent gesture vectors and compared them to the transcript and audio embeddings. We realized that the contrastive loss helps the distribution of these vectors to be close to each other and makes the prediction of them easier for our diffusion model.

Figure 6.1 shows the comparison of two $\mathcal{L}1_{contrastive}$s. At the beginning of the training, the first latent gesture vectors were concentrated in the middle around $(0, 0)$, and $\mathcal{L}1_{contrastive}$ is relatively higher than epoch 321. The higher loss value shows that positive pairs are not close to each other and negative pairs are not far from themselves and we can see that in the visualization as well. When the latent gesture vectors spread more and got closer to the transcript embeddings the $\mathcal{L}1_{contrastive}$ was also decreased.

We can also see the different $\mathcal{L}2_{contrastive}$ and its relation with the distribution of second latent gesture vectors and audio embeddings. The higher the $\mathcal{L}2_{contrastive}$ is the lower the similarity of these embeddings.

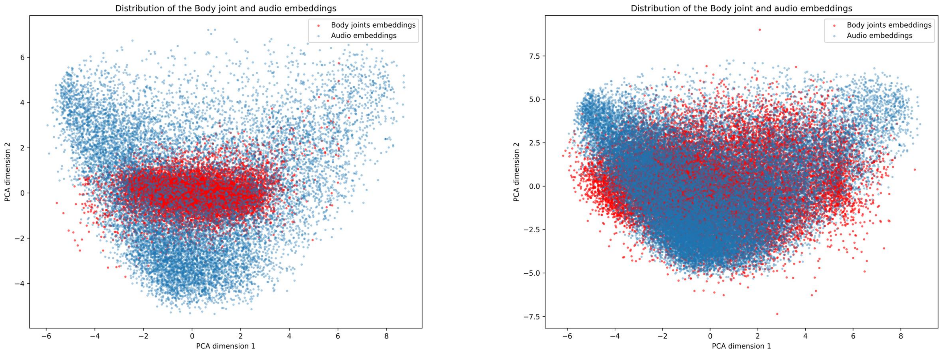
Two points should be mentioned about the contrastive losses here. First,



Epoch 1
$L1_{contrastive} = 7.5093$

Epoch 321
$L1_{contrastive} = 7.2102$

Figure 6.1: Visualization of the distribution of first latent gesture vectors and transcript embeddings before and after training using PCA on the validation set.



Epoch 1
$L2_{contrastive} = 7.8324$

Epoch 321
$L2_{contrastive} = 7.3834$

Figure 6.2: Visualization of the distribution of second latent gesture vectors and audio embeddings before and after training using PCA on the validation set.

the difference between the density of the left samples to the right samples in figure 6.2 and 6.1 is because of the data distributed parallel training on the left samples. DDP is a way of training one model on multiple GPUs to speed up the process. In this method, training samples are divided between the GPUs, and after each epoch, the results from all the GPUs are aggregated together.

Second, there are some scenarios where the embeddings of two modalities look closer to each other, but the contrastive loss shows a higher value. This is a result of the emphasis of this loss on being contrastive rather than only focusing on the similarity. So, the reason for the higher loss could be because of the close distance of the negative pairs.

6.1.2 Contrastive loss

After finishing the implementation of all modules, we ran an experiment to see the effect of each contrastive loss. We first trained two default models with all the features that we introduced for body and hand, then disabled the $\mathcal{L}1_{contrastive}$, $\mathcal{L}2_{contrastive}$, $\mathcal{L}3_{contrastive}$, and $\mathcal{L}_{contrastive}$ individually to measure the impact of each, none, and all of them at the same time, on the results.

We trained the models on the BiGe dataset. For the VQ-VAE models, we trained them for 500 epochs with the initial learning rate of $2e-4$, batch size of 1024, and the learning scheduler on epochs $[100, 200, 300, 400]$ with a gamma 0.5. We set the number of residual blocks to three, and the depth of the Resnet1D in there to six. For the codebook, we considered 512 indices with the length of 512 and for the EMA, we set the μ to 0.99. For the reconstruction loss, we used l1 smooth, and the β of contrastive loss was considered equal to $5e-7$. The temperature of the contrastive loss τ of the audio and transcript was set to 0.01, and for the style, we set it to one.

These values are coming from the many tries and errors we made during this research. But, as training and testing different values for each of these for a certain amount of epoch to compare scientifically requires so much time and resources, we considered that for future works.

These settings were also considered for the other variations, and the comparison is based on the best FGD we observed during these 500 epochs on the validation set. The variations in this experiment are: without $\mathcal{L}1_{contrastive}$, without $\mathcal{L}2_{contrastive}$, without $\mathcal{L}3_{contrastive}$, and without $\mathcal{L}_{contrastive}$ where in each of them we set the value of that loss to zero and we do not use that for backpropagation.

In table 6.1, we can see the results of our experiment on the absence of contrastive loss functions. As it can be seen, our default model achieved the best performance regarding the FGD, $\mathcal{L}1_{contrastive}$, $\mathcal{L}2_{contrastive}$, and $\mathcal{L}3_{contrastive}$ with excluding all the contrastive losses led to the worst performance. These

Exp	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1\downarrow$	$\mathcal{L}2\downarrow$	$\mathcal{L}3\downarrow$
w/o $\mathcal{L}1_{contrastive}$	0.103	5.762	50.27	388.05	None	7.64	226.75
w/o $\mathcal{L}2_{contrastive}$	0.104	5.66	52.52	382.86	7.48	None	229.89
w/o $\mathcal{L}3_{contrastive}$	0.123	5.95	52.46	356.72	7.49	7.67	None
w/o $\mathcal{L}_{contrastive}$	0.12	6.05	43.96	366.62	None	None	None
default	0.09	5.81	50.76	310.44	7.2	7.38	89.27

Table 6.1: Impact of different contrastive losses on our VQ-VAE body model trained on the BiGe dataset. Exp, FGD, MAE, Div, and PPL are experiment name, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

results highlight the importance of contrastive learning in maintaining diversity and accuracy in the training of the VQ-VAE model for gesture generation.

The lowest perplexity in the default model shows confidence in our model in its predictions. While the perplexity in the absence of $\mathcal{L}1_{contrastive}$ is the highest that shows the importance of text modality in the confidence of our VQ-VAE body model.

Exp	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1\downarrow$	$\mathcal{L}2\downarrow$	$\mathcal{L}3\downarrow$
w/o $\mathcal{L}1_{contrastive}$	0.72	10.79	45	391.61	None	8.52	228.34
w/o $\mathcal{L}2_{contrastive}$	0.71	10.74	45.72	396.05	7.47	None	242.68
w/o $\mathcal{L}3_{contrastive}$	0.72	10.91	51.05	385.51	7.47	7.65	None
w/o $\mathcal{L}_{contrastive}$	0.73	10.8	47.33	396.1	None	None	None
default	0.75	10.99	48.87	391.90	7.47	7.72	243.72

Table 6.2: Impact of different contrastive losses on our VQ-VAE hand model. Exp, FGD, MAE, Div, and PPL are experiment name, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

We did the same experiment to generate hand gestures and got different results that can be seen in table 6.2. Based on the table, $\mathcal{L}3_{contrastive}$ has the highest impact on the results where excluding that resulted in the highest diversity, and lowest perplexity, $\mathcal{L}1_{contrastive}$, and $\mathcal{L}2_{contrastive}$. It is probably because of the noise coming from the pose-estimation algorithm on the hand joints.

Also, excluding all the contrastive losses led to slightly better performance based on the FGD and MAE than the default model, but misalignment of latent gesture vectors with the audio, transcript, and style will lead to worse accuracy in the predictions of the diffusion model. Because our diffusion model is guided by audio, transcript, and seed pose, so predicting similar latent vectors is easier.

After training the VQ-VAE models for hand and body, they were used to train the latent diffusion models. We trained each of the variations for 200 epochs with a validation every 20 epochs and saved the best checkpoint monitoring the lowest FGD. The validation of diffusion models takes so much time, so to speed up the process in our limited time we increased the interval. Otherwise, it is not recommended.

The initial learning rate was considered equal to $2e-4$, and batch size equal to 1024, with the learning scheduler on 100th epoch and gamma equal to 0.5. The number of diffusion steps was set to 100 with a linear beta scheduler.

Exp	FGD↓	MAE↓	Div↑
w/o $\mathcal{L}1_{contrastive}$	7.29	24.40	53.63
w/o $\mathcal{L}2_{contrastive}$	4.97	22.42	52.21
w/o $\mathcal{L}3_{contrastive}$	9.69	24.39	55.22
default	4.32	21.43	54.21

Table 6.3: Impact of different contrastive losses on our diffusion body model trained on the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

Table 6.3 shows the results of the best checkpoint after training our diffusion model for 200 epochs on the VQ-VAE body models of previous experience. From this table, we can conclude the importance of contrastive guidance in training the VQ-VAE body model as our default model showed the best performance regarding the FGD, MAE, and diversity.

Table 6.4 shows the results of the best checkpoint after training our diffusion model for 200 epochs on the VQ-VAE hand models of previous experience. From this table, we can conclude the importance of contrastive guidance in training the VQ-VAE hand model as our default model showed

Exp	FGD↓	MAE↓	Div↑
w/o $\mathcal{L}1_{contrastive}$	6.93	28.78	45.67
w/o $\mathcal{L}2_{contrastive}$	5.72	28.62	52.51
w/o $\mathcal{L}3_{contrastive}$	5.73	29.63	48.63
default	5.26	28.43	52.65

Table 6.4: Impact of different contrastive losses on our diffusion hand model trained on the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

the best performance regarding the FGD, MAE, and diversity. However, in comparison to the body model, the experiment on the hand model showed less significant difference and the performance is worse in general than the body model.

6.1.3 Normalization

We also did another experiment to see the effect of normalization on the latent gesture vectors. In this case, we excluded the normalization layer and value normalization of the outputs of our encoder and tested its impact on the reconstruction of the VQ-VAE and prediction of the diffusion model. During this experiment, we realized that although the performance of our VQ-VAE increases predicting the latent gesture vectors in infinite numbers is much harder for our diffusion model and the overall performance is worse than when we normalize the latent gesture vectors.

For this experience, we ran the diffusion training for 500 epochs for our default model with and without the normalization.

Exp	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1$ ↓	$\mathcal{L}2$ ↓	$\mathcal{L}3$ ↓
w/o norm	0.06	5.57	50.07	327.82	7.5	7.74	200
default	0.09	5.81	50.76	310.44	7.2	7.38	89.27

Table 6.5: Impact of normalization on our VQ-VAE body model trained on the BiGe dataset. Exp, FGD, MAE, Div, and PPL are experiment name, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

In table 6.5, 6.6, 6.7, and 6.8 we can observe that despite better FGD results in the VQ-VAE models without normalization, the final results could

Exp	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1\downarrow$	$\mathcal{L}2\downarrow$	$\mathcal{L}3\downarrow$
w/o norm	0.48	10.53	51.17	329.57	7.49	7.68	208.4
default	0.75	10.99	48.87	391.90	7.47	7.72	243.72

Table 6.6: Impact of normalization on our VQ-VAE hand model trained on the BiGe dataset. Exp, FGD, MAE, Div, and PPL are experiment name, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

Exp	FGD↓	MAE↓	Div↑
w/o norm	5.46	17.04	44.03
default	2.82	19.24	55.36

Table 6.7: Impact of normalization on our diffusion body model trained on the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

Exp	FGD↓	MAE↓	Div↑
w/o norm	6.79	27.84	46.3
default	4.73	27.25	51.67

Table 6.8: Impact of normalization on our diffusion hand model trained on the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

be worse as a result of larger domain that our diffusion model should predict. These results were even worse when we did the same experiment on the TED expressive dataset.

6.1.4 EMA

Based on the previous finding in training the VQ-VAE models, we used EMA to update the codebook and set it equal to 0.99 like the Zhang et al. [80] T2M-GPT model. As a result, the codebook is mainly relying on the previous findings and less likely to change suddenly by new values. However, after analyzing the utilization of the codebook, we realized that some indices are used much more than others, and some indices are used very unlikely. So, we put the μ variable in the test to see whether a lower value could help the utilization of the codebook and increase the performance of our models or not.

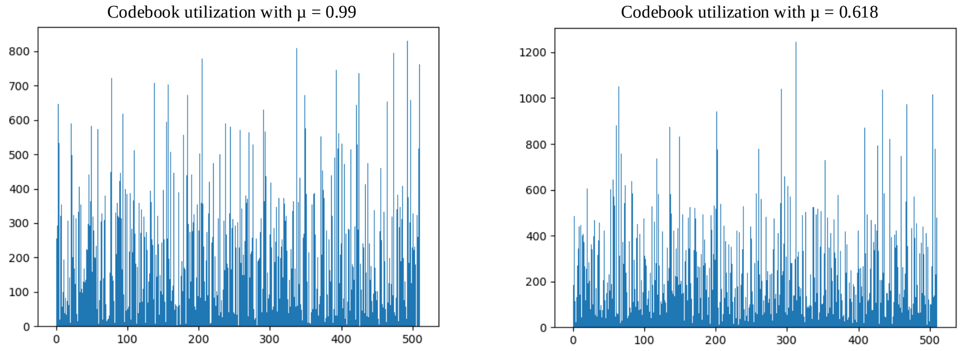


Figure 6.3: Hand mode codebook utilization with $\mu = 0.99$ and 0.618. We can see fewer spikes when $\mu = 0.618$ which resembles better utilization of the codebook.

For testing the effect of μ, the VQ-VAE model was trained for body and hand with the same setup except for the μ value which was considered equal to 0.618.

Exp	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1\downarrow$	$\mathcal{L}2\downarrow$	$\mathcal{L}3\downarrow$
μ = 0.618	0.11	5.6	50.71	391.86	7.47	8.02	212.64
default	0.09	5.81	50.76	310.44	7.2	7.38	89.27

Table 6.9: Impact of decreasing the μ value on training of our VQ-VAE body model on the BiGe dataset. Exp, FGD, MAE, Div, and PPL are experiment name, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

After training the VQ-VAE body model with a lower μ value, we did not observe a significant improvement in the metrics, and the results even turned a bit worse regarding the contrastive losses and perplexity. So, we did not continue training a diffusion model for the body based on the modified VQ-VAE model.

Exp	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1\downarrow$	$\mathcal{L}2\downarrow$	$\mathcal{L}3\downarrow$
μ = 0.618	0.76	10.68	51.37	404.63	7.46	7.67	220.06
default	0.75	10.99	48.87	391.90	7.47	7.72	243.72

Table 6.10: Impact of decreasing the μ value on training of our VQ-VAE hand model trained on the BiGe dataset. Exp, FGD, MAE, Div, and PPL are experiment name, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

In contrast to the body, after modification on the μ value, the metrics showed better results for our VQ-VAE hand model. The contrastive losses were decreased by this modification and the diversity was increased. So, we trained a diffusion model based on this modification for our hand model.

Exp	FGD↓	MAE↓	Div↑
$\mu = 0.618$	4.17	27.38	47.56
default	4.73	27.25	51.67

Table 6.11: Impact of decreasing the μ value on our diffusion hand model trained on the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

After training the diffusion model on top of the modified VQ-VAE hand model, we saw that the FGD and diversity values were decreased. The fall of the FGD value is positive and we thought the drop in the diversity value could be a result of fewer unwanted unnatural movements that generated hands sometimes have. So, this model was chosen over the default model for the evaluation.

6.1.5 Codebook size

We ran another experiment to see the effect of codebook size on our model. Our idea was that fewer codebook indices could help our diffusion model predict the right ones as the probability of randomly choosing them increases. However, there is a limit on decreasing the size of the codebook cause the VQ-VAE model itself should be able to recreate the gestures from this codebook as well.

So, to test the effect of codebook size, we trained our VQ-VAE body model for 100 epochs with different numbers of codebook indices. The results can be seen in the table 6.12. Although we decreased the number of indices, the FGD did not increase much. Also, except the $\mathcal{L}3 \downarrow$, the $\mathcal{L}1 \downarrow$ and $\mathcal{L}2 \downarrow$ did not change so much either. More importantly, the perplexity showed a correlation with the number of codebook indices means the lower the number of codebook indices the higher the confidence of our VQ-VAE model.

As a result, we tried to test our diffusion model with the output of the model with 128 codebook indices as well.

NrCB	FGD↓	MAE↓	Div↑	PPL↓	$\mathcal{L}1 \downarrow$	$\mathcal{L}2 \downarrow$	$\mathcal{L}3 \downarrow$
128	0.16	6.03	53.7	85.34	7.36	7.50	inf
256	0.12	5.34	52.84	163.30	7.36	7.48	inf
512	0.11	5.772	54.41	304.07	7.20	7.39	88.68
1024	20.71	21.67	45.34	807.25	7.46	7.69	247.74

Table 6.12: Impact of decreasing the number of codebook indices on the training of our VQ-VAE body model on the BiGe dataset. NrCB, FGD, MAE, Div, and PPL are the number of codebook indices, freshet gesture distance, mean absolute error on the latent space, diversity, and perplexity respectively.

Our diffusion model with fewer codebook indices showed better results than our default model with the 512 codebook indices regarding the FGD and MAE. This result proves our hypothesis that fewer codebook indices can be beneficial as the diffusion model has a better chance of choosing the right indices.

NrCB	FGD↓	MAE↓	Div↑
128	1.87	18.48	52.78
512	2.82	19.24	55.36

Table 6.13: Impact of decreasing the number of codebook indices on our diffusion body model trained on the BiGe dataset. NrCB, FGD, MAE, and Div are the number of codebook indices, freshet gesture distance, mean absolute error on the latent space, and diversity.

Unfortunately due to the time limitation, we could not continue this experiment on the hand model and update our complete model with the results coming from this experiment. So, the final model is still the default model with the 512 codebook indices.

6.2 Benchmarking

For benchmarking, the default VQ-VAE body model and VQ-VAE hand model with the $\mu = 0.618$ were chosen for our diffusion training. We continued the training until the validation loss converged, and selected the best model based on the lowest FGD. The validation loss over training the diffusion body and hand model can be seen in the figure 6.4 and 6.5. Also, the best validation results are presented in the table 6.14.

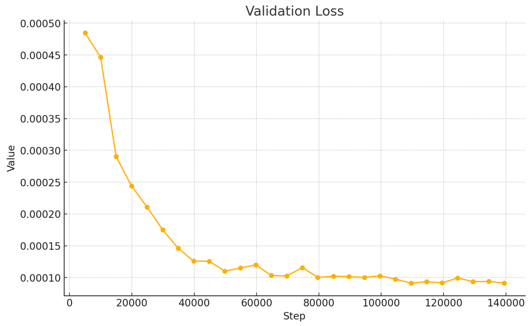


Figure 6.4: Validation loss during the training of diffusion body model on the BiGe dataset.

As it can be seen in the figure 6.4 the diffusion body model was trained smoothly and the validation loss decreased gradually. However, the diffusion hand model was struggling a bit and some fluctuations can be seen in its validation loss in figure 6.5.

Exp	FGD↓	MAE↓	Div↑
body	1.93	15.78	49.22
hand	4.65	27.03	50.07

Table 6.14: Best validation metrics of training the diffusion body and hand model on the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

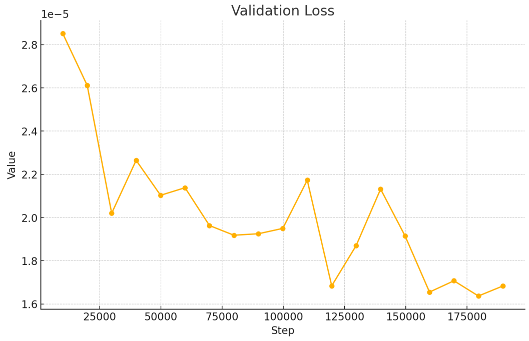


Figure 6.5: Validation loss during the training of diffusion hand model on the BiGe dataset.

6.2.1 DiffGesture

For benchmarking, we chose Zhu et al. [7] DiffGesture model because of three reasons. First, it has been published almost recently and it uses diffusion models. Second, it has been used for benchmarking by other recent papers that show its reproducibility and importance in this field of research. Third, this model only uses audio as input and we can observe the effect of the transcript on the semantic alignment by comparing our model to their model.

So, we updated their public source code to support the BiGe dataset and trained their model for 500 epochs, with the same configuration that they proposed in their GitHub repository. The learning rate of $5e-4$, hidden size 300, latent dimension 128, and diffusion hidden dimension of 512 were used in this training.

6.2.2 Quantitative evaluation

We ran a quantitative evaluation on both models based on the test set of the BiGe dataset. The results can be seen in the table 6.15. Despite we used our default model which was not our best model based on the experiments, still our model showed better results compared to the DiffGesture model. Higher diversity in the DiffGesture model is a result of the noise and unnatural gestures that this model produces after training on the BiGe dataset.

Exp	FGD↓	MAE↓	Div↑
DiffGesture	6.86	43.62	63.76
ours	5.10	33.46	53.93

Table 6.15: Quantitative evaluation on the test set of the BiGe dataset. Exp, FGD, MAE, and Div are experiment name, freshet gesture distance, mean absolute error on the latent space, and diversity.

6.2.3 Qualitative evaluation

As long as there is no proper quantitative metric to evaluate the naturalness and semantic alignment of the gestures, usually, a qualitative evaluation should be done to examine the performance of a method. For this means, we generated some videos by our model and DiffGesture from the test set of the BiGe dataset.

When the results are close together, a survey can be done to ask people about the quality of the generated results. However, the results from the DiffGesture model are marginally worse than our method in this experience. There is so much noise in their results and the human posture does not look natural most of the time. So, we can say for sure that our results achieved better performance than DiffGesture in this case.

Some pictures from the generated videos have been provided in figure 6.6. As we can see, our model can generate gestures that are semantically aligned with the words in the speech and they are close to the ground truth. However, DiffGesture generated completely different gestures that sometimes are not even natural.

Regarding the comparison with the ground truth, the results from our model do not conclude high-frequency movements and the body moves slowly from one gesture to another. This could be a result of generating the rhythm align and semantic align gestures at the same time. Because the semantic align gesture would not happen all along the video and when there are no semantic cues the model tends to generate mean direction vectors to reduce the error.

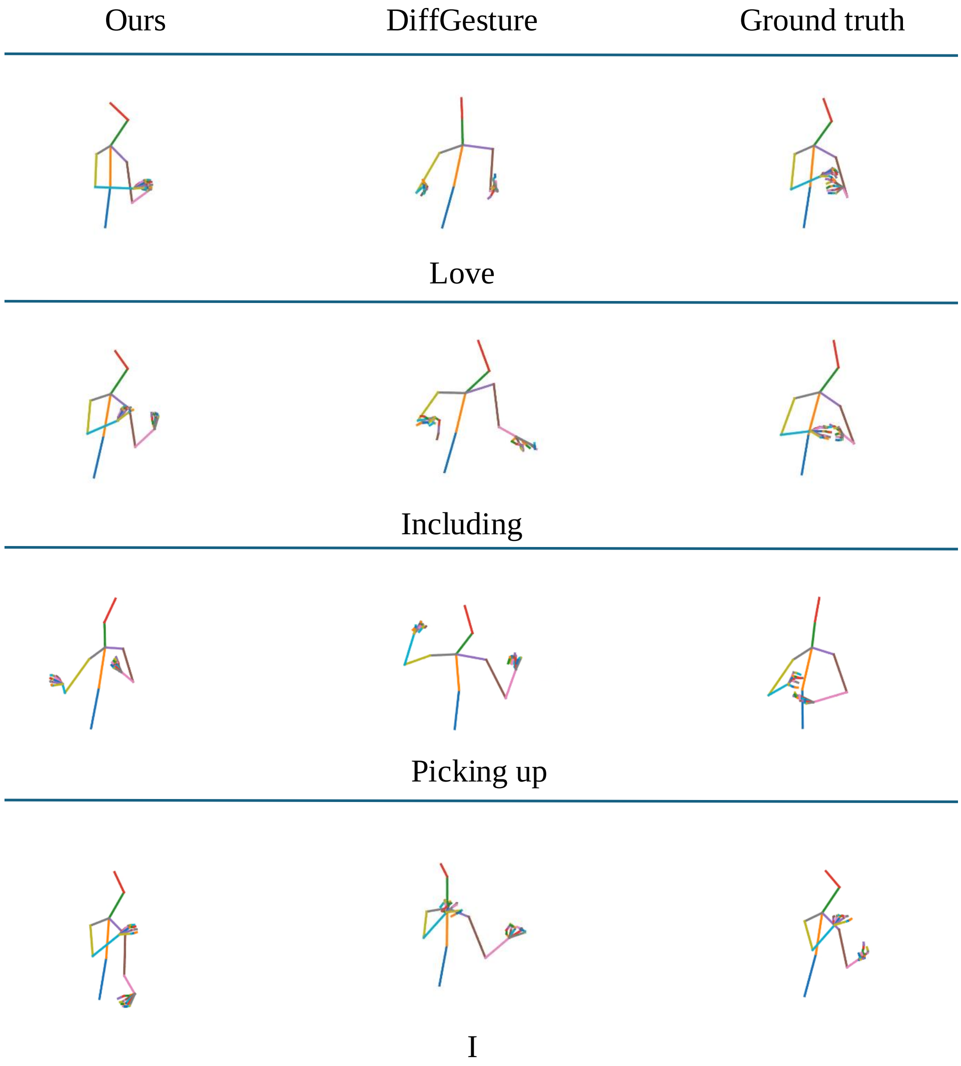


Figure 6.6: The comparison of our model to DiffGesture for generating semantically aligned gestures

7. Conclusion

In this thesis, we explored the challenges and advancements in co-speech gesture generation using deep learning methods. Our approach focused on addressing the semantic grounding problem to improve the human likeness and contextual relevance of generated gestures. Through extensive experimentation and analysis, we developed models that demonstrate improvements in generating semantically aligned gestures.

Our contributions include showing the effectiveness of contrastive learning in the domain of co-speech gesture generation. While previous researchers used contrastive learning in their co-speech gesture generations, as far as we know, no published paper has used contrastive learning to find cues in audio, transcripts, and especially style simultaneously.

Also, we pointed out the difference in the modeling of hand and body joints and the different characteristics they show during gesture generation. While we started this research one VQ-VAE and diffusion model was considered for the whole body, after doing the data analysis we found out about the different distributions of the movements of these parts and decided to separate their training. Further experiments showed the benefits of this decision and opened new opportunities for optimizing the model for that specific part.

Furthermore, we realized that current metrics might not be suitable for evaluating the generated gestures. The high number of joints in the hand makes the FGD metric so sensitive to this part, while the other body parts have more influence on the naturalness and context alignment of the gestures.

In conclusion, our research has made meaningful progress in the field of co-speech gesture generation. The developed models offer a solution for generating natural and contextually relevant gestures, which can be applied in various domains such as digital characters, humanoid robots, and human-computer interactions. Future work can build on these findings by exploring more about different codebooks for rhythm-aligned and semantic-aligned gestures, refining the models, and exploring the learning of similarity measures.

Bibliography

[1] D. Mcneill, "Hand and mind: What gestures reveal about thought," *Bibliovault OAI Repository, the University of Chicago Press*, vol. 27, Jun. 1994. DOI: 10.2307/1576015.

[2] T. Ao, Q. Gao, Y. Lou, B. Chen, and L. Liu, "Rhythmic gesticulator: Rhythm-aware co-speech gesture synthesis with hierarchical neural embeddings," *ACM Transactions on Graphics*, vol. 41, no. 6, pp. 1–19, Nov. 2022, ISSN: 1557-7368. DOI: 10.1145/3550454.3555435. [Online]. Available: http://dx.doi.org/10.1145/3550454.3555435.

[3] X. Liu, Q. Wu, H. Zhou, *et al.*, *Learning hierarchical cross-modal association for co-speech gesture generation*, 2022. arXiv: 2203.13161 [cs.CV].

[4] Y. Liang, Q. Feng, L. Zhu, L. Hu, P. Pan, and Y. Yang, "Seeg: Semantic energized co-speech gesture generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 10 473–10 482.

[5] S. Yang, Z. Wu, M. Li, *et al.*, *Diffusestylegesture: Stylized audio-driven co-speech gesture generation with diffusion models*, 2023. arXiv: 2305.04919 [cs.HC].

[6] Y. Yoon, B. Cha, J.-H. Lee, *et al.*, "Speech gesture generation from the trimodal context of text, audio, and speaker identity," *ACM Transactions on Graphics*, vol. 39, no. 6, pp. 1–16, Nov. 2020, ISSN: 1557-7368. DOI: 10.1145/3414685.3417838. [Online]. Available: http://dx.doi.org/10.1145/3414685.3417838.

[7] L. Zhu, X. Liu, X. Liu, R. Qian, Z. Liu, and L. Yu, *Taming diffusion models for audio-driven co-speech gesture generation*, 2023. arXiv: 2303.09119 [cs.CV].

[8] K. Pang, D. Qin, Y. Fan, *et al.*, "Bodyformer: Semantics-guided 3d body gesture synthesis with transformer," *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–12, Jul. 2023, ISSN: 1557-7368. DOI: 10.1145/3592456. [Online]. Available: http://dx.doi.org/10.1145/3592456.

[9] G. Lee, Z. Deng, S. Ma, T. Shiratori, S. Srinivasa, and Y. Sheikh, "Talking with hands 16.2m: A large-scale dataset of synchronized body-finger motion and audio for conversational motion analysis and synthesis," in *Proceedings of (ICCV) International Conference on Computer Vision*, Oct. 2019, pp. 763–772.

[10] H. Liu, Z. Zhu, N. Iwamoto, *et al.*, "Beat: A large-scale semantic and emotional multi-modal dataset for conversational gestures synthesis," in *European Conference on Computer Vision*, Springer, 2022, pp. 612–630.

[11] S. Nyatsanga, T. Kucherenko, C. Ahuja, G. E. Henter, and M. Neff, "A comprehensive review of data-driven co-speech gesture generation," *Computer Graphics Forum*, vol. 42, no. 2, pp. 569–596, May 2023, ISSN: 1467-8659. DOI: `10.1111/cgf.14776`. [Online]. Available: `http://dx.doi.org/10.1111/cgf.14776`.

[12] Y. Ferstl and R. McDonnell, "Investigating the use of recurrent motion modelling for speech gesture generation," Nov. 2018, pp. 93–98. DOI: `10.1145/3267851.3267898`.

[13] M. Neff, N. Toothman, R. Bowmani, J. E. Fox Tree, and M. A. Walker, "Don't scratch! self-adaptors reflect emotional stability," in *Intelligent Virtual Agents*, H. H. Vilhjálmsson, S. Kopp, S. Marsella, and K. R. Thórisson, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 398–411, ISBN: 978-3-642-23974-8.

[14] M. Kipp, "Gesture generation by imitation: From human behavior to computer character animation," Jan. 2004.

[15] N. Oralbayeva, A. Aly, A. Sandygulova, and T. Belpaeme, "Data-driven communicative behaviour generation: A survey," *ACM Transactions on Human-Robot Interaction*, Aug. 2023. DOI: `10.1145/3609235`.

[16] S. Alexanderson, R. Nagy, J. Beskow, and G. E. Henter, "Listen, denoise, action! audio-driven motion synthesis with diffusion models," *ACM Transactions on Graphics*, vol. 42, no. 4, pp. 1–20, Jul. 2023, ISSN: 1557-7368. DOI: `10.1145/3592458`. [Online]. Available: `http://dx.doi.org/10.1145/3592458`.

[17] A. Deichler, S. Mehta, S. Alexanderson, and J. Beskow, "Diffusion-based co-speech gesture generation using joint text and audio representation," in *INTERNATIONAL CONFERENCE ON MULTIMODAL INTERACTION*, ser. ICMI '23, ACM, Oct. 2023. DOI: `10.1145/3577190.3616117`. [Online]. Available: `http://dx.doi.org/10.1145/3577190.3616117`.

[18] J. Cassell, C. Pelachaud, N. Badler, *et al.*, "Animated conversation: Rule-based generation of facial expression, gesture spoken intonation for multiple conversational agents," *Technical Reports (CIS)*, Nov. 1998. DOI: `10.1145/192161.192272`.

[19] P. Cunningham and S. J. Delany, "K-nearest neighbour classifiers - a tutorial," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–25, Jul. 2021, ISSN: 1557-7341. DOI: `10.1145/3459665`. [Online]. Available: `http://dx.doi.org/10.1145/3459665`.

[20] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, Second Edition, in progress. Cambridge, Massachusetts, London, England: MIT Press, 2018.

[21] C. Zhou, T. Bian, and K. Chen, "Gesturemaster: Graph-based speech-driven gesture generation," Nov. 2022, pp. 764–770. DOI: `10.1145/3536221.3558063`.

[22] H. Voß and S. Kopp, *Augmented co-speech gesture generation: Including form and meaning features to guide learning-based gesture synthesis*, 2023. arXiv: `2307.09597 [cs.HC]`.

[23] Y. Zhi, X. Cun, X. Chen, *et al.*, *Livelyspeaker: Towards semantic-aware co-speech gesture generation*, 2023. arXiv: `2309.09294 [cs.CV]`.

[24] D. Hasegawa, N. KANEKO, S. Shirakawa, H. Sakuta, and K. Sumi, "Evaluation of speech-to-gesture generation using bi-directional lstm network," Nov. 2018. DOI: `10.1145/3267851.3267878`.

[25] S. Wang, S. Alexanderson, J. Gustafson, J. Beskow, G. E. Henter, and É. Székely, "Integrated speech and gesture synthesis," in *Proceedings of the 2021 International Conference on Multimodal Interaction*, ser. ICMI '21, ACM, Oct. 2021. DOI: `10.1145/3462244.3479914`. [Online]. Available: `http://dx.doi.org/10.1145/3462244.3479914`.

[26] T. Ao, Q. Gao, Y. Lou, B. Chen, and L. Liu, "Rhythmic gesticulator: Rhythm-aware co-speech gesture synthesis with hierarchical neural embeddings," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 6, pp. 1–19, 2022.

[27] U. Bhattacharya, E. Childs, N. Rewkowski, and D. Manocha, "Speech2affectivegestures: Synthesizing co-speech gestures with generative adversarial affective expression learning," in *Proceedings of the 29th ACM International Conference on Multimedia*, ser. MM '21, ACM, Oct. 2021. DOI: `10.1145/3474085.3475223`. [Online]. Available: `http://dx.doi.org/10.1145/3474085.3475223`.

[28] I. Habibie, W. Xu, D. Mehta, *et al.*, *Learning speech-driven 3d conversational gestures from video*, 2021. arXiv: `2102.06837 [cs.CV]`.

[29] T. Kucherenko, D. Hasegawa, G. E. Henter, N. Kaneko, and H. Kjellström, "Analyzing input and output representations for speech-driven gesture generation," in *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*, ser. IVA '19, ACM, Jul. 2019. DOI: `10.1145/3308532.3329472`. [Online]. Available: `http://dx.doi.org/10.1145/3308532.3329472`.

[30] J. Li, D. Kang, W. Pei, *et al.*, *Audio2gestures: Generating diverse gestures from speech audio with conditional variational autoencoders*, 2021. arXiv: `2108.06720 [cs.CV]`.

[31] S. Taylor, J. Windle, D. Greenwood, and I. Matthews, "Speech-driven conversational agents using conditional flow-vaes," Dec. 2021, pp. 1–9. DOI: `10.1145/3485441.3485647`.

[32] S. Ghorbani, Y. Ferstl, D. Holden, N. F. Troje, and M.-A. Carbonneau, *Zeroeggs: Zero-shot example-based gesture generation from speech*, 2022. arXiv: `2209.07556 [cs.GR]`.

[33] C. Ahuja, D. W. Lee, R. Ishii, and L.-P. Morency, "No gestures left behind: Learning relationships between spoken language and freeform gestures," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 1884–1895. DOI: `10.`

18653/v1/2020.findings-emnlp.170. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.170.

[34] Z. Zhang, T. Ao, Y. Zhang, *et al.*, *Semantic gesticulator: Semantics-aware co-speech gesture synthesis*, 2024. arXiv: 2405.09814 [cs.GR]. [Online]. Available: https://arxiv.org/abs/2405.09814.

[35] T. Ao, Z. Zhang, and L. Liu, *Gesturediffuclip: Gesture diffusion model with clip latents*, 2023. arXiv: 2303.14613 [cs.CV].

[36] J. Chen, Y. Liu, J. Wang, A. Zeng, Y. Li, and Q. Chen, *Diffsheg: A diffusion-based approach for real-time speech-driven holistic 3d expression and gesture generation*, 2024. arXiv: 2401.04747 [cs.SD].

[37] M. H. Mughal, R. Dabral, I. Habibie, L. Donatelli, M. Habermann, and C. Theobalt, *Convofusion: Multi-modal conversational diffusion for co-speech gesture synthesis*, 2024. arXiv: 2403.17936 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2403.17936.

[38] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132 306, Mar. 2020, ISSN: 0167-2789. DOI: 10.1016/j.physd.2019.132306. [Online]. Available: http://dx.doi.org/10.1016/j.physd.2019.132306.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, Dec. 1997. DOI: 10.1162/neco.1997.9.8.1735.

[40] E. Shlizerman, L. Dery, H. Schoen, and I. Kemelmacher-Shlizerman, "Audio to body dynamics," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Jun. 2018. DOI: 10.1109/cvpr.2018.00790. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2018.00790.

[41] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, *Generative adversarial networks*, 2014. arXiv: 1406.2661 [stat.ML].

[42] D. Bank, N. Koenigstein, and R. Giryes, *Autoencoders*, 2021. arXiv: 2003.05991 [cs.LG].

[43] H. Yi, H. Liang, Y. Liu, *et al.*, *Generating holistic 3d human motion from speech*, 2023. arXiv: 2212.04420 [cs.CV].

[44] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, *Neural discrete representation learning*, 2018. arXiv: 1711.00937 [cs.LG].

[45] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2023. arXiv: 1706.03762 [cs.CL].

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

[47] J. Ho, A. Jain, and P. Abbeel, *Denoising diffusion probabilistic models*, 2020. arXiv: 2006.11239 [cs.LG].

[48] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, *Hierarchical text-conditional image generation with clip latents*, 2022. arXiv: 2204.06125 [cs.CV].

[49] K. Mei and V. M. Patel, *Vidm: Video implicit diffusion models*, 2022. arXiv: 2212.00235 [cs.CV].

[50] A. Gulati, J. Qin, C.-C. Chiu, *et al.*, *Conformer: Convolution-augmented transformer for speech recognition*, 2020. arXiv: 2005.08100 [eess.AS].

[51] A. Radford, J. W. Kim, C. Hallacy, *et al.*, *Learning transferable visual models from natural language supervision*, 2021. arXiv: 2103.00020 [cs.CV].

[52] A. Rahamim and Y. Belinkov, *Contrasim – a similarity measure based on contrastive learning*, 2023. arXiv: 2303.16992 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2303.16992.

[53] T. Kucherenko, R. Nagy, M. Neff, H. Kjellström, and G. E. Henter, *Multimodal analysis of the predictability of hand-gesture properties*, 2022. arXiv: 2108.05762 [cs.HC].

[54] S. Alexanderson and G. E. Henter, "Robust model training and generalisation with Studentising flows," in *Proceedings of the ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, ser. INNF+'20, vol. 2, 2020, 25:1–25:9. [Online]. Available: https://arxiv.org/abs/2006.06599.

[55] J. M. Joyce, "Kullback-leibler divergence," in *International Encyclopedia of Statistical Science*, Springer, 2011, pp. 720–722.

[56] X. Huang and S. Belongie, *Arbitrary style transfer in real-time with adaptive instance normalization*, 2017. arXiv: 1703.06868 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1703.06868.

[57] B. Jiang, X. Chen, W. Liu, J. Yu, G. Yu, and T. Chen, *Motiongpt: Human motion as a foreign language*, 2023. arXiv: 2306.14795 [cs.CV].

[58] L. Ji, P. Wei, Y. Ren, J. Liu, C. Zhang, and X. Yin, *C2g2: Controllable co-speech gesture generation with latent diffusion model*, 2023. arXiv: 2308.15016 [cs.CV].

[59] S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, and J. Malik, *Learning individual styles of conversational gesture*, 2019. arXiv: 1906.04160 [cs.CV].

[60] E. Volkova, S. de la Rosa, H. Bülthoff, and B. Mohler, "The mpi emotional body expressions database for narrative scenarios," *PloS one*, vol. 9, e113647, Dec. 2014. DOI: 10.1371/journal.pone.0113647.

[61] K. Takeuchi, D. Hasegawa, S. Shirakawa, N. Kaneko, H. Sakuta, and K. Sumi, "Speech-to-gesture generation: A challenge in deep learning approach with bi-directional lstm," Oct. 2017, pp. 365–369. DOI: 10.1145/3125739.3132594.

[62] Y. Yoon, W.-R. Ko, M. Jang, J. Lee, J. Kim, and G. Lee, *Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots*, 2018. arXiv: 1810.12541 [cs.RO].

[63] K. Takeuchi, S. Kubota, K. Suzuki, D. Hasegawa, and H. Sakuta, "Creating a gesture-speech dataset for speech-based automatic gesture generation," May 2017, pp. 198–202, ISBN: 978-3-319-58749-3. DOI: 10.1007/978-3-319-58750-9_28.

[64] T. Kucherenko, P. Jonell, Y. Yoon, P. Wolfert, and G. E. Henter, "A large, crowdsourced evaluation of gesture generation systems on common data: The genea challenge 2020," in *26th International Con-*

ference on Intelligent User Interfaces, ser. IUI '21, ACM, Apr. 2021. DOI: 10.1145/3397481.3450692. [Online]. Available: http://dx.doi.org/10.1145/3397481.3450692.

[65] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, *Openpose: Realtime multi-person 2d pose estimation using part affinity fields*, 2019. arXiv: 1812.08008 [cs.CV].

[66] R. Ochshorn and M. Hawkins, *GitHub - lowerquality/gentle: Gentle forced aligner — github.com*, https://github.com/lowerquality/gentle, [Accessed 11-03-2024], 2016.

[67] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, *3d human pose estimation in video with temporal convolutions and semi-supervised training*, 2019. arXiv: 1811.11742 [cs.CV].

[68] V. Choutas, G. Pavlakos, T. Bolkart, D. Tzionas, and M. J. Black, *Monocular expressive body regression through body-driven attention*, 2020. arXiv: 2008.09062 [cs.CV].

[69] H.-S. Fang, J. Li, H. Tang, *et al.*, *Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time*, 2022. arXiv: 2211.03375 [cs.CV].

[70] F. Zhang, V. Bazarevsky, A. Vakunov, *et al.*, *Mediapipe hands: On-device real-time hand tracking*, 2020. arXiv: 2006.10214 [cs.CV].

[71] F. Zhang, X. Zhu, and M. Ye, *Fast human pose estimation*, 2019. arXiv: 1811.05419 [cs.CV].

[72] A. Lücking, K. Bergmann, F. Hahn, S. Kopp, and H. Rieser, "The bielefeld speech and gesture alignment corpus (saga)," Jan. 2010. DOI: 10.13140/2.1.4216.1922.

[73] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2019. arXiv: 1810.04805 [cs.CL].

[74] M. E. Peters, M. Neumann, M. Iyyer, *et al.*, *Deep contextualized word representations*, 2018. arXiv: 1802.05365 [cs.CL].

[75] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, *Wav2vec 2.0: A framework for self-supervised learning of speech representations*, 2020. arXiv: 2006.11477 [cs.CL].

[76] W.-N. Hsu, B. Bolte, Y.-H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. PP, pp. 1–1, Oct. 2021. DOI: 10.1109/TASLP.2021.3122291.

[77] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, *Data2vec: A general framework for self-supervised learning in speech, vision and language*, 2022. arXiv: 2202.03555 [cs.LG].

[78] R. Li, S. Yang, D. A. Ross, and A. Kanazawa, *Ai choreographer: Music conditioned 3d dance generation with aist++*, 2021. arXiv: 2101.08779 [cs.CV].

[79] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, *Gans trained by a two time-scale update rule converge to a local nash equilibrium*, 2018. arXiv: 1706.08500 [cs.LG].

[80] J. Zhang, Y. Zhang, X. Cun, *et al.*, *T2m-gpt: Generating human motion from textual descriptions with discrete representations*, 2023. arXiv: 2301.06052 [cs.CV].

[81] M. Rebol, C. Gütl, and K. Pietroszek, "Passing a non-verbal turing test: Evaluating gesture animations generated from speech," in *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, IEEE, Mar. 2021. DOI: 10.1109/vr50410.2021.00082. [Online]. Available: http://dx.doi.org/10.1109/VR50410.2021.00082.

[82] H. Voß and S. Kopp, *Aq-gt: A temporally aligned and quantized gru-transformer for co-speech gesture synthesis*, 2023. arXiv: 2305.01241 [cs.HC].

[83] X. Chen, B. Jiang, W. Liu, *et al.*, *Executing your commands via motion diffusion in latent space*, 2023. arXiv: 2212.04048 [cs.CV].

[84] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, *High-resolution image synthesis with latent diffusion models*, 2022. arXiv: 2112.10752 [cs.CV].

[85] O. Ronneberger, P. Fischer, and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, 2015. arXiv: 1505.04597 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1505.04597.