POLITECNICO DI TORINO

Master's Degree in Computer Engineering

Master's Degree Thesis

Animating Virtual Characters in Unity Using Generative AI: A Prompt-Based Approach

Supervisors

Prof. Andrea BOTTINO

Prof. Francesco STRADA

Dott. Stefano CALZOLARI

Candidate

Ciro ANNICCHIARICO

APRIL 2025

Abstract

Creating realistic and expressive character animations is a major challenge in video game development and interactive applications. Traditional methods, such as keyframing and motion capture, demand considerable time and resources. Recently, AI-driven text-to-motion models, especially diffusion models, have emerged as a promising alternative, allowing for the automatic creation of animations from textual descriptions.

This thesis offers a comparative analysis of different motion generation models, focusing primarily on diffusion-based techniques. The evaluation examines crucial factors like motion quality, realism, and adherence to prompts. To connect AI-generated animations with game engines, a specialized tool was developed to enable the seamless integration of these models into Unity, providing a practical workflow for developers.

In addition to technical evaluation, this work explores whether text-to-motion models can effectively express emotions through movement. By drawing on insights from body language research, a structured approach was created to enhance motion prompts for generating emotionally expressive animations. An experiment was conducted where participants identified emotions in AI-generated animations. The responses were analyzed to evaluate the strengths and weaknesses of these models in producing believable emotional expressions.

Finally, this thesis points out significant limitations of prompt-based AI models, such as the quality and diversity of training datasets, which greatly affect the expressiveness and generalization of the generated animations. The study concludes with suggestions for future research directions and improvements to enhance the adaptability and quality of AI-generated animations for upcoming applications in gaming and interactive media.

"You miss 100% of the shots you don't take.
Wayne Gretzky"
Michael Scott

Table of Contents

List of Tables

List of Figures

Acronyms

AI
Artificial Intelligence

VAE
Variational Autoencoder

VR
Virtual Reality

MC
Motion Capture

NLP
Natural Language Processing

ETMG
Emotion-enriched Text-to-Motion Generation

XR
Extended Reality

VH
Virtual Human

IK
Inverse Kinematics

NPC
Non-Player Character

Chapter 1

Introduction

Animating virtual characters convincingly is a central and enduring challenge within the fields of video game development and interactive media. Traditional animation methods, such as keyframing and motion capture, while effective, often require extensive manual labor, specialized expertise, and substantial resources. These approaches are frequently impractical or prohibitively expensive, particularly for independent developers or projects with constrained timelines. In recent years, generative artificial intelligence (AI) has emerged as a transformative solution, offering powerful tools capable of synthesizing realistic human motion from textual descriptions alone.

Among these AI-driven methods, diffusion models have shown particular promise. These models generate detailed and diverse animations by iteratively refining random noise into coherent, lifelike movements, bridging the gap between ease of use and animation quality. Nonetheless, integrating these AI-generated motions into environments, such as game engines like Unity, presents its own set of technical and practical challenges. Ensuring seamless output compatibility demands careful consideration.

This thesis explores the potential of different text-to-motion generative models for animating virtual characters within Unity, addressing both technical implementation and the expressive capabilities of AI-driven animations. A dedicated Unity-based tool has been developed to facilitate easy integration and utilization of state-of-the-art generative models, enabling the direct synthesis and application of animations from simple textual prompts.

Furthermore, this work investigates an often-overlooked aspect of generative motion synthesis: emotional expressiveness. While generative models can create realistic animations, their ability to convey nuanced emotions through movement remains unclear. By drawing upon established frameworks from body language research this thesis introduces structured prompt engineering approaches aimed at enhancing the emotional clarity and expressiveness of AI-generated animations.

Through experimental evaluations involving human participants, this study systematically assesses the ability of current generative models to produce emotionally believable and prompt-adherent animations. Preliminary results of the experiments indicate that certain emotions, such as sadness and anger, can be effectively conveyed through full-body animations alone. In contrast, other emotions, like surprise and disgust, pose greater challenges due to the absence of nuanced expressive details, such as facial expressions and detailed hand gestures. Additionally, the evaluation revealed significant differences in performance among models, highlighting that some text-to-motion models are more effective than others in producing emotionally recognizable animations.

These insights underline the strengths and current limitations of text-to-motion models, pointing toward promising directions for future advancements in emotionally expressive animation generation.

Part of the work presented in this thesis has been selected for presentation at the **XR Salento 2025-Internation Conference on eXtended Reality** and will contribute to an upcoming scientific publication. This reflects the relevance and applicability of the research conducted.

Chapter 2

Background

Generative Artificial Intelligence (AI) is a rapidly evolving field focused on creating new content, such as text, images, and even videos, through machine learning models. In contrast to traditional AI systems built for classification or predictive tasks, generative models analyze patterns and structures from their training data to create completely new realistic outputs that exhibit similar characteristics.

The foundation of generative AI lies in probabilistic modeling, where neural networks are trained to understand and replicate the underlying distribution of data. These models can interpolate between known data points, generate variations of existing content, and even create completely new samples based on their training. Over the past decade, advances in deep learning have pushed generative AI forward, enabling applications across various domains, such as natural language processing (NLP), computer vision, game development, and animation.

In recent years, AI has significantly advanced the field of motion synthesis, leading to the development of various generative models that generate human motion from various inputs. The main architectures include Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), Normalizing Flows, Transformer-based architectures, and, more recently, Diffusion Models. Each of these approaches offers distinct advantages and trade-offs regarding realism, diversity, and controllability of generated motion.

This chapter examines the architectural aspects of each approach and provides insights into their methodologies for motion synthesis. Additionally, it gives a first look at the current state-of-the-art in generating expressive animations for virtual humans (VHs), highlighting existing limitations and outlining the aim of this study.

2.1 Generative AI Architectures

2.1.1 Generative Adversarial Networks

Generative Adversarial Networks (GAN) [1] are based on an adversarial learning process in which two neural networks, the generator and the discriminator,compete against each other in a zero-sum game (see Fig. 2.1):

- The generator is responsible for creating synthetic data that resemble real-world samples.
- The discriminator attempts to distinguish between the real data from the training set and the synthetic data produced by the generator.

Through iterative training, the generator continually improves by producing outputs that become increasingly indistinguishable from real data, while the discriminator refines its ability to differentiate real from fake content. Eventually, the adversarial process reaches a balance where the generator produces highly realistic samples.

Despite their success, GANs are known to suffer from mode collapse, where the generator learns to produce only a limited variety of outputs instead of capturing the full diversity of the training data. Furthermore, training GANs is computationally intensive and requires careful balancing to ensure both networks improve at comparable rates.

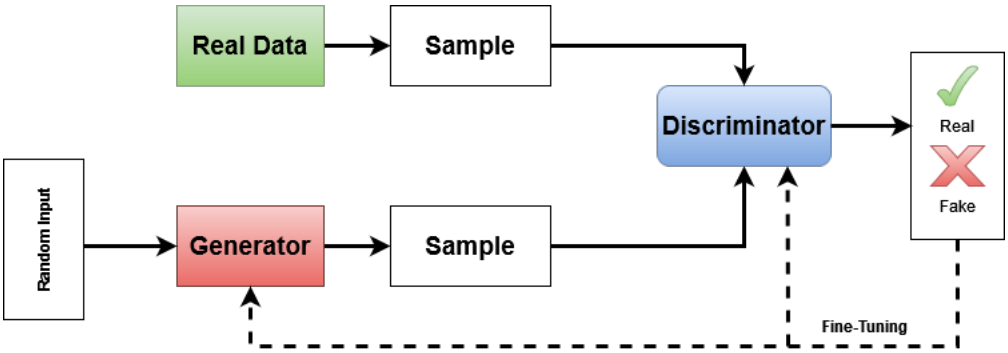


Figure 2.1: GAN: Generator and discriminator

2.1.2 Transformers

Transformers arrived as a milestone in generative AI and ushered in a whole spectrum of tasks, including machine translation and language modeling [2]. Their design has been one of the most influential works in the history of neural network

architecture, mainly because of the pioneering use of attention mechanisms, which have become one of the most crucial elements in sequence-to-sequence processing.

Transformers can learn complicated relations and patterns in data sequences by self-attention and multi-head attention, irrespective of the distance between the elements. In this way, transformers can realize many relations and patterns in the input.

This is the typical use case for transformers in NLP, where positional encoding is applied to the input sequence, allowing the model to retain the order of words and maintain context. Thus, transformers lend themselves particularly to being used in the construction of powerful generative models, such as Generative Pre-trained Transformers (GPT), which can produce coherent and contextually relevant text.

Models like OpenAI's GPT and BERT (Bidirectional Encoder Representations from Transformers) exemplify the transformative impact of this architectural approach on generative AI and NLP.

2.1.3 Variational Autoencoders

Another important generative AI model is the Variational Autoencoder [3]. VAEs consist of two major parts: an encoder and a decoder. An encoder compresses the input into lower-dimensional space—what is referred to as the latent space—and the decoder reconstructs this compressed form back into the original input form (see Fig.2.2).

One of the most defining characteristics of VAEs is introducing variation into the latent space by mapping the data to a standard Gaussian distribution. This allows the model to generate outputs with the same mean and variance as the original input, thereby creating realistic samples with coherent features.

By learning compact, meaningful representations of the data, VAEs can do more than just reproduce the input data; they generate new samples from scratch by following the learned distribution. This makes VAEs a potentially very useful class of algorithms for tasks in image generation, where generating outputs with similar characteristics to those of the training data is heavily desired.

2.1.4 Diffusion models

Diffusion models, introduced by [5], represent a recent and powerful approach in the field of generative artificial intelligence. They are designed to generate data by gradually refining random noise into coherent and realistic outputs.

These models operate through a two-step process, namely a forward process and a reverse process [6]. In the forward process, data (such as an image or motion sequence) is progressively transformed into noise by applying small amounts of Gaussian noise over a series of steps, effectively 'diffusing' the data into a noisy

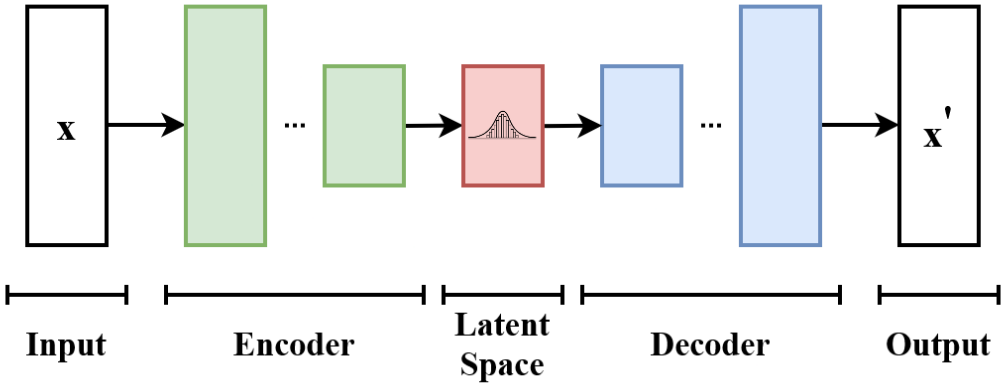


Figure 2.2: The basic scheme of a variational autoencoder. The model receives x as input. The encoder compresses it into the latent space. The decoder receives as input the information sampled from the latent space and produces x' as similar as possible to x [4].

representation. In the reverse process, the model learns to reverse this noisy state step-by-step, ultimately recovering a realistic, noise-free sample that resembles the training data (see Fig.2.3).

The principal strength of diffusion models is their capacity to generate high-quality outputs with fine details. In contrast to GANs, which can exhibit instability and mode collapse (resulting in limited diversity), diffusion models are more stable and generate diverse samples with minimal artifacts.

Furthermore, diffusion models possess inherent control over the generative process, allowing for flexible sampling at various stages. This has made them particularly effective for applications requiring fine-grained detail, such as image and motion generation, where realistic textures, smoothness, and nuanced features are essential.

Moreover, diffusion models have demonstrated adaptability across a range of domains, including natural language processing, audio synthesis, and motion animation. This versatility makes them a valuable tool in the field of generative AI. Their ability to produce detailed, high-fidelity results has established diffusion models as a leading approach in state-of-the-art generative techniques.

2.2 Generative AI for Motion Synthesis

A key distinction between different architectures is how they internally represent motion. The two main approaches are:

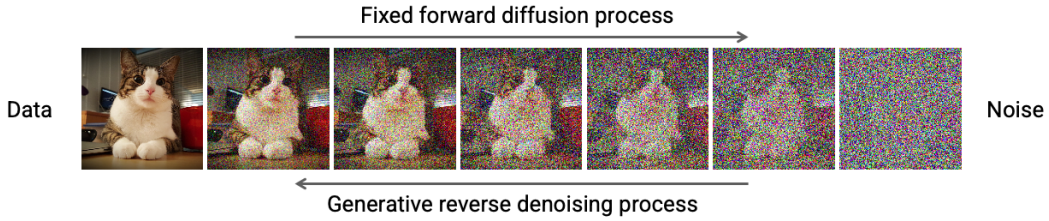


Figure 2.3: Diffusion model processes moving to and from data and noise [7]

1. **Continuous Representations**: the model learns a latent space where motion sequences are encoded as continuous vectors. These vectors capture high-level motion features and are typically used in autoencoder-based architectures.

2. **Discrete Representations**: motion sequences are transformed into tokenized representations, with each frame or segment mapped to a limited set of learned motion *codes*. This approach is often seen in vector quantization methods.

The choice of representation significantly affects how models generate and refine motion sequences.

Autoencoder-based architectures focus on compressing and reconstructing motion sequences. These models are made up of two primary components: an encoder, which maps an input motion sequence into a lower-dimensional latent space, and a decoder which reconstructs a motion sequence from the latent space. The latent space can be structured as either deterministic (standard autoencoders) or probabilistic (variational autoencoders, VAEs). Deterministic Autoencoders learn a direct mapping between motion and its latent encoding, aiming to optimize reconstruction accuracy while Variational Autoencoders (VAEs) introduce a probabilistic latent space, which allows for the generation of diverse motion sequences by sampling various points in the distribution. A significant challenge in autoencoder-based methods is ensuring that the latent space captures enough semantic information from text while also maintaining high-frequency motion details.

Autoregressive models create motion by generating each frame one at a time, using the previously created frames as a basis for the next. These models are generally implemented through Recurrent Neural Networks (RNNs) which analyze motion sequences frame by frame, effectively capturing short-term dependencies, or Transformers which utilize self-attention mechanisms to understand long-range dependencies, which enhances their ability to manage complex motions. Autoregressive models handle data in a sequential manner, which can lead to the accumulation of errors over time (known as exposure bias). During training, a method called

teacher forcing is often used, where the model learns from actual sequences, but during inference, it must depend on its own predictions.

On the other hand, diffusion models approach motion generation as a process of iterative refinement. These models begin with a sequence of random noise and gradually denoise it to produce a coherent motion output. A noise function is applied to the original motion data across several steps, which degrades the information while a neural network is trained to reverse this degradation, reconstructing a believable motion sequence. Unlike autoregressive models, diffusion models generate all frames at once, thus avoiding exposure bias. However, they do require a significant number of inference steps, which can make them computationally intensive.

In terms of animation quality, each architecture has strengths and weaknesses regarding motion realism, diversity, and controllability. VAE-based methods provide good diversity but often fall short on fine details. GAN-based techniques can create realistic animations, yet they are difficult to train and tend to lack diversity. Transformer-based models strike a balance between realism and controllability, although they encounter issues with exposure bias. On the other hand, diffusion models excel in both motion realism and diversity, positioning them as the most promising options for text-driven motion generation.

2.3 Expressive Animations

The use of generative AI techniques for motion synthesis, as discussed in the previous section, has the potential to greatly simplify and accelerate the animation pipeline. This is particularly valuable in the development of applications involving Virtual Humans (VHs), where large quantities of human-like motion are often required. Automatically generating realistic and varied motion can reduce the manual effort involved in animating characters, making it easier to prototype and scale interactive scenarios, especially in XR environments where motion diversity and believability play a crucial role in enhancing immersion.

2.3.1 Why VHs?

VHs are digital agents with anthropomorphic features widely used in entertainment, gaming, education, therapy, training, and customer service. Their ability to simulate human-like behaviors and emotions significantly enhances user immersion and engagement, making them crucial in applications requiring believable human-computer interactions (HCI) or realistic crowds in extended reality (XR) scenarios.

Emotional expressiveness of Virtual Humans is key to their success as it plays a critical role in facilitating credible interaction that generates empathy, social presence, and a stronger sense of being in virtual environments. High-profile applications include virtual therapists capable of expressing true empathy to achieve therapeutic

Figure 2.4: Examples of Virtual Humans [8, 9, 10]

results [9], emotionally expressive virtual agents that add realism to simulated public service training scenarios [11], interactive NPCs that add storytelling depth and player immersion in gaming [12], and very realistic virtual civilians employed in complicated peacekeeping operations or evacuation scenarios to gain insights into crowd dynamics and human reactions under critical circumstances [10]. Thus, the integration of emotionally expressive and socially attentive virtual humans is needed to promote more realistic, effective, and compelling interactions in virtual worlds. This approach enhances user satisfaction and performance in a range of applied domains.

2.3.2 Emotional Body Language

The communication of emotions in virtual humans relies on three main channels: facial expression, body language, and prosody. While Virtual Humans research has focused heavily on facial expressions [13] and vocal prosody [14], body language remains an equally important but less researched aspect of emotional expression in

VHs [15].

Body language is a powerful means of nonverbal communication that complements or even replaces facial expressions in conveying emotions. According to [16], body language plays an important role in recognizing emotions because it is visible from a distance, can express emotions when facial expressions are not available, and is cross-cultural. Certain postures and movements are consistently associated with specific emotions. For example, pride is often expressed through an upright posture and an expanded chest, while sadness is manifested in slumped shoulders and a hunched posture. Other studies also confirm that body movements alone can significantly influence the perception of emotions [17, 18], underscoring their importance in non-verbal communication.

To better understand how emotions are conveyed through body movements, researchers have developed structured frameworks such as the Body Action and Posture Coding System (BAP) [19] and the Laban Movement Analysis (LMA) [20, 21]. These systems do not generate animations but provide valuable tools for analyzing and categorizing expressive body language.

BAP

The BAP is a structured method designed for the detailed and precise coding of body movements, particularly in the studies of emotional expression. The BAP system addresses a significant gap in nonverbal behavior research, where consensus on a common, detailed approach to coding body movements, particularly those related to emotions, was lacking.

BAP codes movement at anatomical, form, and functional levels. At an anatomical level, it identifies movements in terms of body parts such as head, trunk, arms, and legs. Form-level coding specifies directions and orientations of movement in three-dimensional space relative to an anatomical reference. Functionally, BAP distinguishes movements as emblems (culturally defined gestures), illustrators (speech support), and manipulators (self-regulatory physical interactions).

BAP concisely identifies action segmentation, i.e., preparation, stroke, and retraction. This increases the precision of timing and sequence analysis. Validation using the Geneva Multimodal Emotion Portrayals (GEMEP) corpus [22] has confirmed its reliability.

LMA

Laban Movement Analysis (LMA), originally developed by Rudolf Laban, provides a structured framework for analyzing human movement in fields such as dance, communication, and therapy. LMA classifies movements into six main categories: Body, Space, Effort, Shape, Relationship, and Phrasing (see Fig.2.6). Each category offers specific parameters for systematically observing movements.

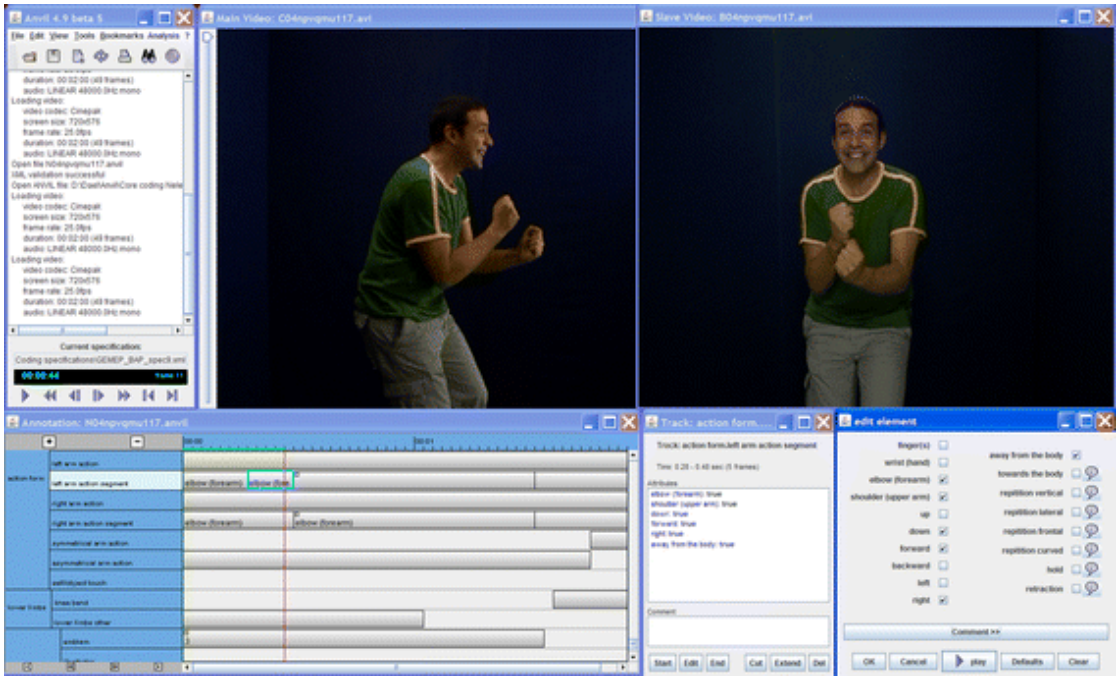


Figure 2.5: BAP coding platform

For example, Effort describes the qualitative energy and dynamics of movements, while Shape analyzes how the form of movements changes in relation to space and relationships. LMA also utilizes notation methods such as Phrase Writing, Motif Writing, and Labanotation, along with coding sheets, to facilitate precise documentation and analysis. This versatility allows observers to capture movements comprehensively at both macro and micro levels.

Limitations

Despite their potential, these frameworks have only been used to a limited extent in animation systems and remain primarily tools for motion analysis rather than motion generation. Some attempts have been made to integrate LMA into interactive and procedural animation frameworks, such as spatial motion doodles [24], which use hand gestures in VR to create expressive character animations, or models such as EMOTE [25], which apply LMA principles to synthesize shape variations in movement. Similarly, efforts have been made to incorporate emotion-driven gestures into embodied conversational agents [26]. However, these approaches remain limited in scope and are not widely used in current animation pipelines.

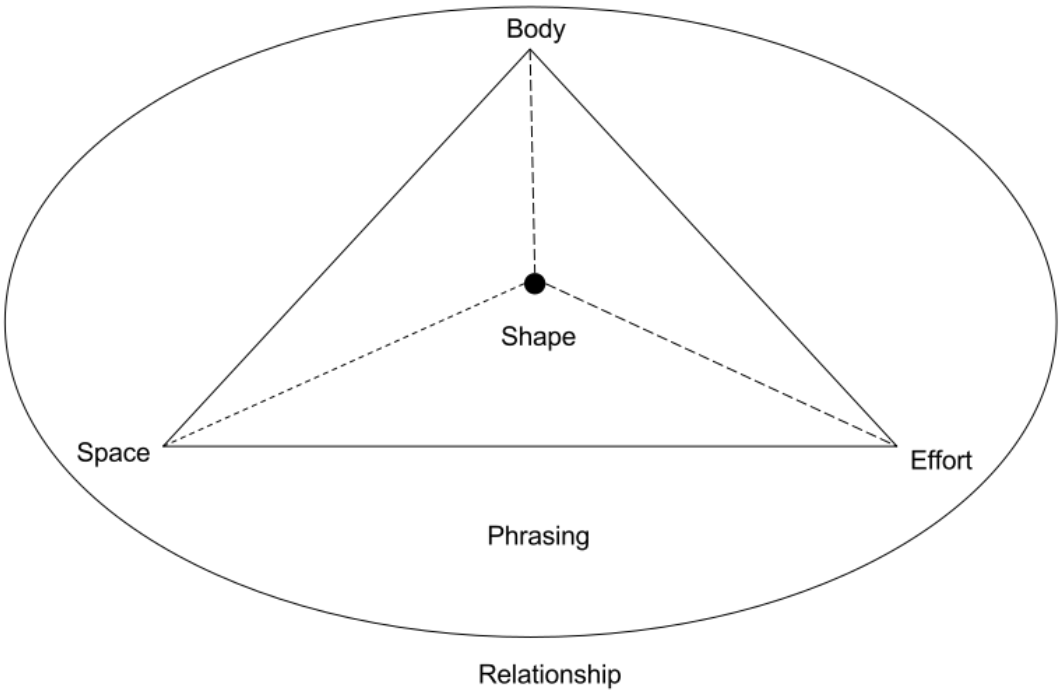


Figure 2.6: LMA categories [23]

2.3.3 Text-to-Motion Animation

As a result, expressive body language in animation is still predominantly generated through traditional animation workflows, where experienced animators manually design the movements based on artistic expertise and psychological research. While this approach can produce high-quality results, it is labor intensive and requires significant artistic and technical skills. Animators must carefully craft each movement to ensure that the character's movements align with the intended emotional state, often relying on observational studies, acting techniques, and psychological insights.

Motion capture (MC) offers an alternative by allowing actors to perform movements that naturally incorporate emotional body language. However, capturing new MC data specifically for emotional performances is often just as time-consuming and resource-intensive as hand-keyed animation. As a result, many studies turn to existing MC datasets, which typically focus on general motion sequences rather than explicitly capturing emotion-driven body language. This reliance on neutral datasets introduces further challenges, since adapting emotion-neutral MC data to create expressive animations often requires significant post-processing and manual refinement to create the necessary emotional depth. This limitation makes it difficult to generate animations that are not only realistic, but also believable in

their emotional intent.

Recent advances in Artificial Intelligence (AI) have opened new possibilities. Modern speech synthesis technologies now allow the procedural generation and real-time modulation of vocal prosody, enabling Virtual Humans to convey varied emotional states dynamically. Similarly, AI-driven motion synthesis, especially latent diffusion models such as MDM [27] and LADiff [28], has made procedural generation of realistic body movements possible from textual descriptions or contextual prompts. However, current AI-driven models often prioritize physical realism and coherent motion rather than emotional expressivity. The few notable exceptions are L^3EM [29], which unfortunately do not investigate emotion believability, and co-speech gesture models like ZeroEGGS [30] or EMoG [31].

Co-speech gesture models have notably advanced in synthesizing expressive upper-body gestures aligned with speech rhythm, prosody, and semantics. However, these models have significant limitations: they typically rely on spoken audio or speech transcripts rather than textual or descriptive prompts, restricting their broader applicability. Furthermore, they focus primarily on upper-body gestures accompanying speech, neglecting the wider range of full-body movements necessary for everyday actions like walking, sitting, or object interactions.

2.4 Research Objectives

Considering the limitations of existing generative models (the lack of emotional believability investigations using text-to-motion, and the restricted focus on co-speech models), this study investigates how effectively advanced text-to-motion models can generate emotionally expressive animations through carefully designed textual prompts.

To achieve this, two preliminary experiments are conducted using state-of-the-art full-body text animation models. The first experiment evaluates the degree to which each animation adheres to the specific instructions detailed in the textual prompts. This step validates the clarity and precision of prompts, ensuring they effectively guide the models in generating animations that accurately reflect intended emotions and actions.

Building on the validated prompts from the first experiment, the second experiment then specifically investigates the effectiveness of advanced text-to-motion models in generating emotionally expressive animations. Both experiments incorporate user studies for a human-centered evaluation, making this research among the first to directly assess the emotional expressiveness of text-generated animations by explicitly prompting users to recognize emotions.

Ultimately, this approach confirms not only that animations convey emotions convincingly but also that they faithfully embody the specific, context-dependent

cues provided in the textual instructions. This foundational analysis supports future research into integrating generative AI with the animation workflows of virtual humans (VHs), particularly aiming to enhance non-verbal communication in immersive applications.

Chapter 3

Related Works

3.1 Text-to-Motion Models

The evolution of text-to-motion models has progressively shifted from deterministic motion mapping to probabilistic diffusion-based approaches. MotionDiffuse [32] introduced this paradigm, where motion sequences emerge through a denoising process. A significant advancement in this area is Motion Latent Diffusion (MLD) [33], which presents a new method by transitioning the diffusion process from raw motion sequences to a more compact latent space (see Fig.3.1). This approach minimizes computational demands while maintaining high-quality motion synthesis. MLD utilizes a variational autoencoder (VAE) to develop a structured latent representation of motion, greatly enhancing efficiency in both training and inference. By taking advantage of this latent space, MLD demonstrates competitive performance across various human motion generation tasks, such as text-to-motion, action-to-motion, and unconditional motion synthesis.

Expanding on these ideas, AttT2M [34] enhances text-driven motion generation by using a multi-perspective attention mechanism. Unlike earlier models that view motion-text relationships as a one-dimensional issue, AttT2M incorporates body-part attention to boost spatial encoding and employs a global-local attention mechanism to strengthen the connection between text and motion (see Fig.3.2). By utilizing VQ-VAE for motion quantization, the model creates a more refined representation, leading to improved semantic alignment and greater control in motion generation.

In parallel, research has increasingly focused on improving the quality and diversity of the generated animations. Models like ReMoDiffuse [35] combines retrieval-based augmentation with the diffusion process. Unlike traditional diffusion models that create motion sequences solely from a learned latent space, ReMoDiffuse uses external motion samples that are retrieved based on both semantic and

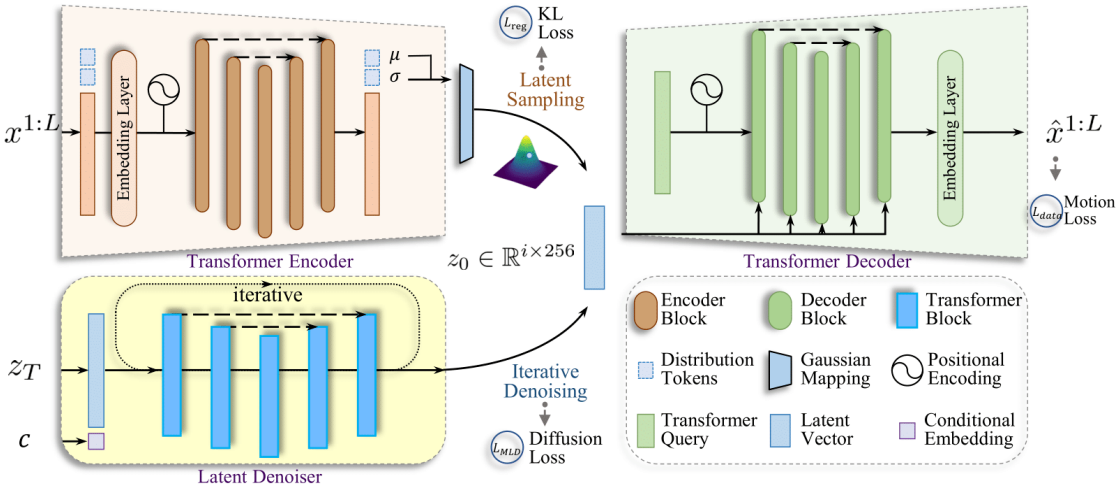


Figure 3.1: MLD Architecture

kinematic similarities. This hybrid retrieval approach enhances the denoising process.

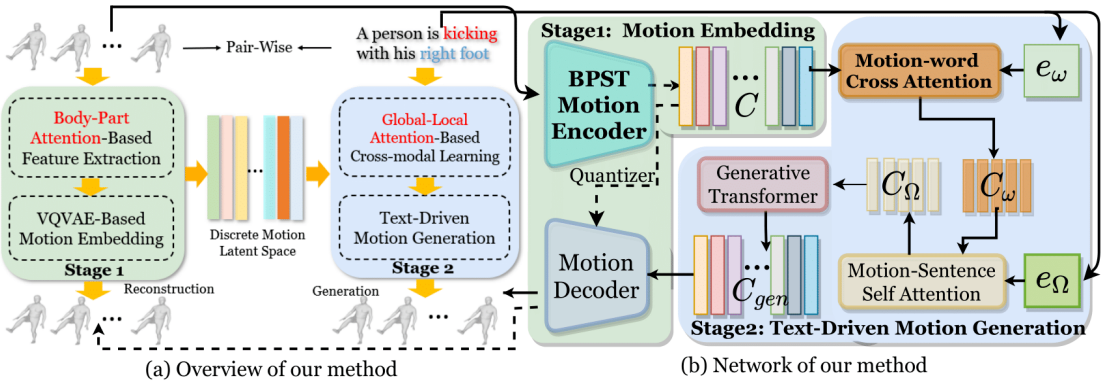


Figure 3.2: AttT2M Architecture

The Human Motion Diffusion Model (MDM) is a transformer-based classifier-free diffusion model (see Fig.3.3) specifically designed for human motion generation that allows the application of geometric losses to motion attributes, thus helping to enforce foot contact constraints [27]. MDM supports various conditioning modes that enable tasks such as text-to-motion and action-to-motion. Its effectiveness as a generative prior for motion synthesis has led to several advancements aimed at enhancing spatial control and efficiency. One notable advancement is Guided Motion Diffusion (GMD) [36], which improves MDM by adding spatial constraints, such as predefined trajectories and obstacle avoidance. GMD features an emphasis projection mechanism to ensure that the generated motion aligns more closely with

spatial information, while also utilizing a dense signal propagation technique to spread sparse keyframe constraints throughout the sequence, allowing for accurate motion trajectories and the synthesis of more realistic motions in applications where the synthesized motions need to follow certain constraints. OmniControl [37] takes a more general approach, enabling spatial control over any joint at any time. Unlike GMD, which focuses primarily on the pelvis trajectory, OmniControl simultaneously incorporates spatial guidance for multiple joints and refines the entire motion sequence for greater realism. Using a combination of spatial and realism guidance, OmniControl dynamically adjusts motion to adhere to control signals while maintaining coherence and natural movement. This makes OmniControl particularly suitable for applications that require fine-grained control, such as ensuring that a hand reaches a specific object or a head maintains a safe distance from an obstacle. To tackle the efficiency challenges of MDM, the Efficient Motion Diffusion Model (EMDM) [38] introduces a new acceleration strategy for motion diffusion models. Traditional diffusion models typically require numerous denoising steps to preserve motion quality, but EMDM incorporates a conditional denoising diffusion GAN, allowing for high-fidelity motion generation with significantly fewer sampling steps. Another significant extension is PriorMDM [39], which utilizes MDM as a generative prior to facilitate composition-based motion synthesis. PriorMDM presents three types of motion composition: sequential composition for creating long, continuous motion sequences, parallel composition for interactions involving multiple characters, and model composition for blending various motion priors. This framework effectively broadens MDM's capabilities beyond generating short-duration motion for a single person, enabling it to handle complex, structured motion synthesis scenarios. Lastly, Diffusion Noise Optimization (DNO) [40] enhances MDM by treating motion editing and refinement as an optimization challenge within the diffusion noise latent space. Rather than retraining a diffusion model for every individual task, DNO adjusts the latent noise during inference through gradient-based optimization. So DNO is model agnostic and can be used with any diffusion model. This approach allows for detailed motion editing, including changes to joint trajectories, pose refinement, and adherence to constraints, all without the need for extra training data or alterations to the base model. These advancements collectively showcase the adaptability of MDM as a foundational model for motion generation.

Beyond spatial control and efficiency, researchers have begun integrating temporal control with Length-Aware Latent Diffusion (LADiff) [28]. Unlike MDM-based models, LADiff addresses the challenge of controlling the duration of synthesized motions by introducing a length-aware variational autoencoder that learns motion representations with length-aware latent codes. Furthermore, a length-conformant latent diffusion model is used to generate motions with a level of detail proportional to the length of the target sequence. This enables the adaptation of motion

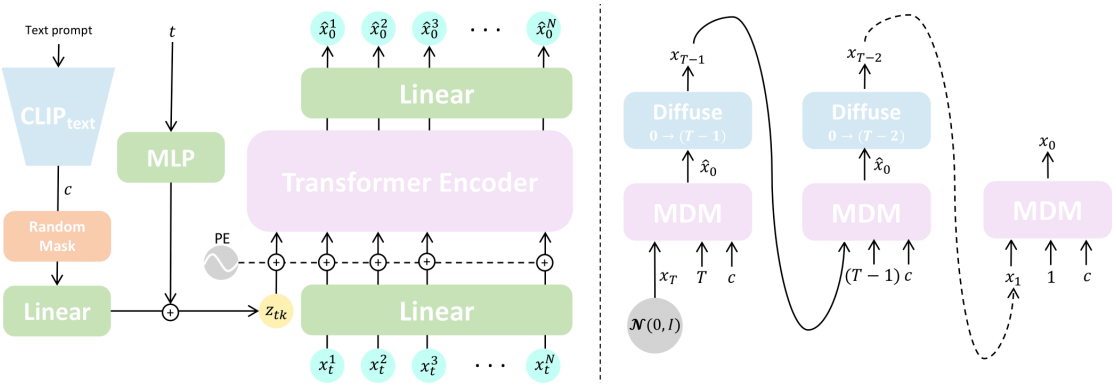


Figure 3.3: MDM Architecture

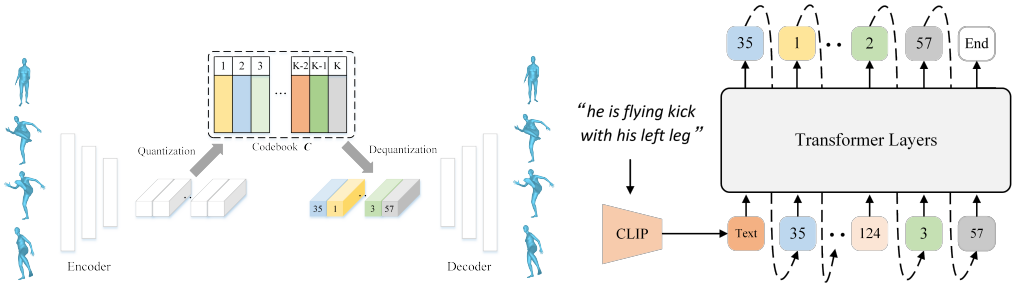
dynamics based on the specified animation duration.

Further extending the capabilities of diffusion-based models, FLAME [41] offers a cohesive method for motion synthesis and editing. Unlike earlier models that were limited to generation, FLAME empowers users to manipulate motion sequences through free-form textual descriptions directly. This capability allows for precise adjustments to motion at both the frame and joint levels without the need for fine-tuning, representing a major advancement in interactive motion design.

Motion decomposition techniques have also emerged as a key area of improvement. LGTM [42], employs a distinctive local-to-global strategy for text-driven motion generation. By utilizing large language models (LLMs) to break down textual descriptions into motions specific to body parts, LGTM achieves better semantic alignment between motion segments and their corresponding textual cues. The inclusion of specialized body-part motion encoders refines local details, while a full-body optimizer maintains coherence in the overall motion sequence. This hierarchical approach addresses the shortcomings of existing text-to-motion methods that often misassign actions to individual body parts. Moreover, MoFusion [43] presents a motion synthesis framework that goes beyond text conditioning by integrating music as an additional input. By aligning generated motions with the rhythmic and structural elements of an audio track, MoFusion creates new opportunities for dance choreography and interactive character animation, showcasing enhanced adaptability across various conditioning inputs.

The challenge of physical awareness has been addressed by PhysDiff [44], which integrates a physics-aware strategy for motion diffusion by ensuring physical plausibility during the denoising process. By integrating a physics-based motion projection module, it significantly enhances realism in generative motion models.

Exploring different paradigms, T2M-GPT [45] adopts a GPT-like framework for text-to-motion generation (see Fig.3.4). This model uses a Vector Quantized Variational AutoEncoder (VQ-VAE, Fig.3.4a) to map motion sequences to discrete

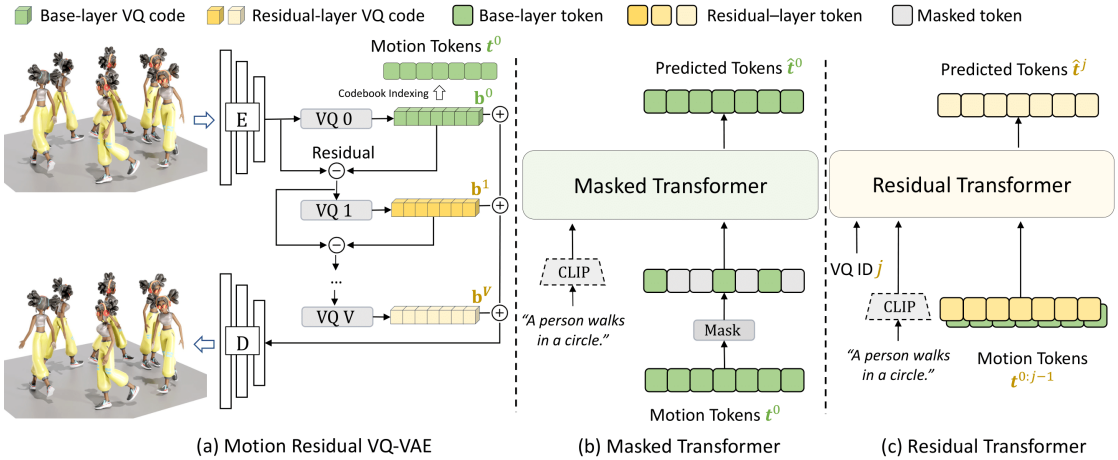


(a) Motion VQ-VAE **(b)** T2M-GPT

Figure 3.4: T2M-GPT Architecture

code indices, followed by a Generative Pre-trained Transformer (GPT, Fig.3.4b) that generates sequences of code indices from pre-trained text embeddings. T2M-GPT has proven to be particularly effective in processing longer and more detailed text prompts, generating complex motion sequences that closely are found more adherent than other models to the given descriptions.

In contrast, MoMask [46] presents a completely new approach to motion generation, drawing inspiration from masked modeling techniques used in image and NLP tasks. Rather than generating motion sequences autoregressively (like T2M-GPT) or through a denoising process (as seen in diffusion models), MoMask iteratively fills in missing motion tokens using a two-stage framework. This includes a hierarchical residual vector quantization (RVQ) system that breaks down motion into several layers of tokens, along with a masked transformer architecture that predicts the missing motion tokens in a non-autoregressive, bidirectional way (see Fig.3.5).



(a) Motion Residual VQ-VAE (b) Masked Transformer (c) Residual Transformer

Figure 3.5: MoMask Architecture

Recent advancements in text-to-motion technology have led to models that enhance generative capabilities by focusing on human-object and multi-character interactions. HOIAnimator [47] and HOI-Diff [48] are designed to create realistic human-object interactions (HOI) by simulating the movements of both humans and objects within a diffusion-based framework. HOIAnimator employs Perceptive Diffusion Models (PDM) to independently model the motions of humans and objects, ensuring smooth interactions through a perceptive message-passing system. In contrast, HOI-Diff utilizes a dual-branch diffusion approach along with an Affordance Prediction Diffusion Model (APDM) to predict likely contact points between humans and objects, thereby improving physical realism. InterControl [49] addresses the challenge of multi-character interactions, enabling the generation of synchronized movements among multiple humans. It features a motion controller and inverse kinematics guidance to accurately position joint locations across different agents. By using a diffusion-based framework, InterControl achieves zero-shot generalization for any number of interacting characters, marking a significant advancement in the creation of realistic group behaviors.

3.2 Limitations

Despite these advancements, current methods predominantly prioritize coherence, physical realism, and motion diversity, leaving explicit emotional expressiveness relatively unexplored. Addressing this gap is crucial for improving the realism and effectiveness of Virtual Human interactions.

Co-speech models such as ZeroEGGS [30], EMoG [31], GestureDiffuCLIP [50], Speech2AffectiveGestures [51] and AMUSE [52] have made notable progress in generating expressive animations linked to actual spoken content. They can integrate stylistic and emotional nuances into the resulting gestures; however, they typically rely on audio or text transcripts from spoken dialogue rather than descriptive prompts, limiting their usability in broader animation scenarios. Moreover, in virtual reality applications requiring virtual humans' crowds, it is often unnecessary to generate co-speech movement. In this context dyadic user-agent interactions do not occur, making co-speech animation worthless while pivoting the focus on agent-to-agent speech animations not associated to any specific spoken text. In this scenarios generic speech gestural animations can still convey emotional significance to the user, in addition to saving computational resources and avoiding tedious animation authoring. Since these models usually focus on upper body gestures that accompany speech, they often lack the ability to produce general body movements such as walking, sitting, or interacting with objects while expressing emotions [53, 54].

Prior research, whether on general text-to-motion models or specific to co-speech

ones, rarely conducts user evaluations focused on the believability of the emotional expressions. Instead, investigations predominantly assess factors such as human likeness, naturalness, appropriateness to the accompanying speech or audio, and how well the generated motion represents a given target style or emotion, explicitly stating it. This inherently limits the assessment of true emotional believability, as users are not independently identifying the emotional expression.

The first text-to-motion model that attempts to incorporate emotion in motion generation is SMooDi (Stylized Motion Diffusion Model) [55], but its primary goal is not to generate emotional animations. Instead, SMooDi focuses on style transfer, adapting a pre-trained text-to-motion model to generate stylized motion by conditioning on both textual descriptions and reference motion sequences. The model is trained on the 100STYLE dataset [56], which predominantly captures movement styles rather than emotional expressions. While the dataset includes a few emotion-related categories, such as angry, depressed, and proud, the majority of its styles are centered around locomotion patterns, personality traits, and movement styles (e.g. airplane pose, zombie or childlike walk). As a result, SMooDi does not generate animations explicitly driven by emotional intent but rather applies style transfer to modify an existing motion sequence to reflect a given stylistic attribute. Another example is [29], which focused on Emotion-enriched Text-to-Motion Generation (ETMG) with L^3EM, an LLM-driven approach that introduces emotional expression at the limb level in full-body animations. Although the method improves motion realism, it has the disadvantage that its evaluation is purely quantitative and it lacks user studies on the believability of the expressed emotions. In addition, the source code is not publicly accessible, which prevents a direct comparison with competing approaches.

Thus, while SMooDi and the L^3EM approach represent a step towards incorporating emotion in motion synthesis, the challenge of generating motion that is both emotionally believable and expressive remains an open research question. In this preliminary study, the aim is to start addressing these limitations, investigating to what extent the current state-of-the-art text-to-animation AI models are capable of generating believable body language for VHs.

Chapter 4

Methods

Traditionally, animating characters involves manual techniques such as keyframing or motion capture, which can be time-consuming and resource-demanding. However, with the rise of AI-driven generative models, it has become possible to automate the creation of realistic animations directly from text descriptions, opening up new opportunities for developers and digital artists.

The first part of the chapter focuses on the experimental setup, outlining the criteria for selecting generative models, the rationale behind designing specific text-to-motion prompts, and the systematic creation of a dataset of animations used for user studies. Then, the implementation of a tool in Unity that allows for the use of text-to-motion models to generate and apply animations within the game engine is outlined. The aim is to establish an efficient integration between Unity and motion generation models, enabling users to create animations from simple text prompts without the need for manual input, offering a flexible and automated tool that enhances the animation workflow for developers and interactive content creators.

Communication between the game engine and the generation system is handled via ZeroMQ, a protocol that facilitates data exchange between Unity and a Python script managing the models. The system's output is then converted into a Unity-compatible format (FBX) to be applied to 3D models through the Humanoid rig.

This chapter aims to provide an in-depth look at the choices made during implementation, the tool's architecture, and the challenges faced throughout development.

4.1 Experiments Setup

In order to generate believable emotional animations, this study systematically compares state-of-the-art AI motion synthesis models evaluating their ability to generate animations that effectively convey a defined emotion and match the given prompts. This study followed a three-stage approach:

1. Generating a dataset of AI-driven animations using refined text-to-motion prompts informed by research on emotion and body language [16].
2. Conducting a first user study to evaluate the prompt affinity of AI-generated animations.
3. Conducting a second user study to evaluate human perception of the emotional expressiveness of AI-generated animations.

Specifically, all six universally recognized basic emotions (*happiness, sadness, anger, fear, surprise, and disgust*) [57], three common actions, and four text-to-motion models were considered. The following sections discuss how the models (Section 4.1.1) and the actions were selected (Section 4.1.2), and explain the prompt design approach (Section 4.1.3) that guides each model toward generating emotionally meaningful motion.

4.1.1 Model selection

In order to conduct an effective evaluation, four text-to-motion generative models have been selected among the state-of-the-art plethora. The model selection phase followed a structured filtering process that combined both theoretical considerations and empirical evaluations. Initially, key criteria were defined to guide model selection based on the objectives of generating emotionally expressive animations:

1. **Public availability**: only models with publicly accessible implementations were considered, facilitating reproducibility and comparative analysis.
2. **Quality assessment**: an initial empirical evaluation was conducted through a visual assessment of animations generated from preliminary prompts to judge motion quality and coherence qualitatively. In particular, models were excluded if they (1) consistently generated physically impossible animations such as joint intersections (e.g., arms intersecting with torso) or impossible rotations (e.g., knees rotated backward to the facing direction); (2) persistently failed to depict the action described in the input prompt.
3. **Architectural diversity**: models representing diverse generative paradigms (diffusion-based and transformer-based approaches) were prioritized to broadly evaluate different technological solutions.

4. **Up-to-date coverage**: preference was given to models that have been released or updated recently, ensuring that the selected solutions reflect current practices within the field.

MotionDiffuse [32] and its sub-model SMooDi [55], as well as MoMask [46], exhibited relatively lower quality in preliminary tests and were therefore excluded from further analysis. Similarly, specialized variants derived from MDM [36, 37, 38, 44] were excluded since preliminary evaluations indicated no significant qualitative improvement over the foundational MDM model. To the best of our knowledge, the unique ETMG-oriented model [29] has no publicly available code by now, thus it was discarded in this evaluation as not compliant with the selection criteria.

The outcome of this filtering process led to the selection of four generative models, each offering distinct advantages aligned with this study's goals:

- **LADiff** [28], a length-aware latent diffusion approach chosen for its ability to handle motions of varying durations. By subdividing the latent space into subspaces specialized for different temporal spans, it provides a controllable framework for modulating sequence length.

- **MDM** [27], a diffusion-based model developed around a transformer architecture, featuring iterative denoising steps that refine noised motion data into coherent animations. Selected as the main foundational diffusion model, which has been continuously updated by the present time as compared to its derivatives.

- **T2M-GPT** [45], a transformer-based model that employs discrete latent codes to process textual prompts but does not implement motion data diffusion techniques. Chosen as designed to handle more detailed and extended descriptions, possibly translating linguistic nuances into expressive and varied motion sequences.

- **Muse Animate** [58], selected primarily due to its direct compatibility with Unity and relevance as the main industry solution, despite the absence of public information about its architecture.

4.1.2 Action Selection

To evaluate each model's capacity to convey emotion in realistic scenarios, three actions that frequently occur in everyday life and XR contexts were carefully selected. This choice not only guarantees practical relevance but also provides distinct movement patterns where emotional variations are readily recognizable and can be systematically studied. The selected actions are:

- **Standing**: often referred to as the "Idle" animation in XR contexts, this action represents minimal movement when a VH is not actively engaging in other explicit tasks. Standing animations are critical to study emotional expressivity as they test the model's capability to convey nuanced emotional states through subtle posture variations and minimal body movements, making it highly relevant in scenarios with limited explicit action [59, 60].

- **Speaking**: unlike co-speech animation, which synchronizes upper-body gestures with speech content, speaking in this study refers to a more general animation depicting a character engaged in speech-related body movements. This approach allows for evaluating full-body emotional expressivity independent of verbal content, making it relevant for VH interactions where speech synchronization is unnecessary, such as background characters in XR environments.

- **Walking**: gait and posture variations during walking have been widely studied in emotion recognition research. Walking provides a clear and consistent context to evaluate how effectively models embed emotion into dynamic, commonly occurring movements [61, 62].

4.1.3 Prompt design

The prompt design process followed an iterative approach to determine the most effective way to generate emotionally expressive animations. Initially, a simple prompt structure that combined only an *emotion* and an *action* was experimented, using the following template:

$$a\ \textbf{\textit{[emotion]}}\ person\ \textbf{\textit{[action]}} \tag{4.1}$$

This format was intended to test whether generative models could interpret abstract emotional descriptors and produce coherent animations accordingly. However, the results varied significantly across models. Among the four models tested, *Muse Animate* (in the following referred as *Muse*) was the only one that consistently produced expressive and recognizable emotional animations using this prompt structure. For example, prompts such as "an angry person is walking" led to animations that exhibited noticeable tension and abrupt movement, suggesting that *Muse* may have been trained with explicitly labeled emotion data or that it employs a mechanism that effectively links textual emotional cues to movement patterns.

In contrast, *LADiff*, *MDM*, and *T2M-GPT* struggled with this prompt format. Their outputs were often neutral, overly generic, or physically realistic but emotionally ambiguous. These models failed to consistently differentiate between emotions,

frequently generating similar animations regardless of the specified emotional state. This inconsistency likely stems from the fact that their training datasets do not contain explicit emotion labels or enough contextual examples associating textual emotion descriptors with distinct movement variations. Furthermore, repeated generations with the same prompt sometimes resulted in vastly different motions, revealing weak emotional consistency while tending to default to animations lacking clear emotional intent.

Recognizing the limitations of the initial prompt template, the approach was refined by incorporating *bodily behavior* descriptors, explicitly defining postures, movement characteristics, and gesture patterns to provide clearer guidance for generating emotionally expressive animations. The new prompt format was structured as follows:

$$\textit{a person } \textbf{\textit{[action]}} \textit{ with } \textbf{\textit{[bodily behavior]}}_1, \ldots, \textit{and } \textbf{\textit{[bodily behavior]}}_n \quad (4.2)$$

The term *bodily behaviors* refers to specific motion cues—such as arm positioning, head inclination, or gait patterns—that are strongly associated with particular emotional expressions. These behaviors were drawn from the table in Witkower et al. [16] (with additional insights from Depraz et al. [63]), which compiles empirical evidence from multiple studies on how postural and gestural features contribute to emotion recognition. The study identifies behaviors that have been consistently validated across different research contexts, providing a structured basis for defining emotional movement patterns. For example, *sadness* is often associated with slumped shoulders and a downward-tilted head, while *anger* is characterized by tense posture and forceful, abrupt limb movements.

Although template 4.2 significantly improved generation results, many animations generated by MDM, LADiff and T2M-GPT were still not able to fully portray the described bodily behaviors. Therefore, these prompts were further refined following an iterative approach until coherence between prompt and motion was achieved (Tab. 4.1). Specifically, refinements included:

- **Sentence structure**: long sentences were divided using punctuation and varying conjunctions (e.g., in (*sadness*, *walking*, *T2M-GPT*) the prompt was rephrased in three sentences).

- **Word synonym**: variations in wording were used where specific nouns or verbs were incorrectly depicted (e.g., in (*happiness*, *speaking*, *T2M-GPT*) "speaking" was replaced with "talking").

- **Verb tense**: when models were unable to portray the specified action correctly verb tenses were adjusted. T2M-GPT in particular often struggled when

using present continuous, then present simple was used instead (e.g., in (*sadness*, *walking*, *T2M-GPT*) present simple is used instead of continuous).

- **Parameter omission**: bodily behaviors or actions that were not effectively substituted by a synonym were omitted. *Standing* action in particular was often omitted (e.g., in ($fear$, *standing*, *MDM*) prompt is missing "standing") as it represents a lack of deliberate activity, thus involving no actual motion.

To empirically validate the resulting prompts, a pre-evaluation survey was submitted (reported in Section 5.2.1), asking participants to quantify the coherence between prompts and the resulting animation clips. Results of prompt-coherence (Section 6.2) were considered satisfying and ratified the prompt design method.

Emotion	Action	Model	Generated Prompt
Happiness	Speaking	T2M-GPT	A person is talking and jumping with their head tilted upward. They use energetic and rhythmic hand gestures swinging freely.
		LADiff	A person is speaking with head tilted up and rhythmic arm movements as if happy.
		MDM	A person is speaking with head tilted up and rhythmic arm movements almost dancing.
		Muse	A happy person is speaking.
Sadness	Walking	T2M-GPT	A person walks with a sluggish pace, their head tilted downward and shoulders slumped. Their arms hang loosely at their sides with little movement. They occasionally drag their feet slightly or look downward as if avoiding eye contact.
		LADiff	A person is walking slowly with head tilted down, collapsed body, and with a little swing of arms.
		MDM	A person is walking slowly with head tilted down.
		Muse	A sad person is walking.
Fear	Idle	T2M-GPT	A person stands nervously, their body slightly hunched and their hands held up near their chest or face.
		LADiff	A person is covering his face with hands collapsing his upper body and arms.
		MDM	A person is covering his face with his hands collapsing his upper body and arms with slow movements.
		Muse	A person is standing in fear.

Table 4.1: Examples of Prompt Design for Different Emotions and Actions.

With this refined prompts, *LADiff* and *MDM* demonstrated a noticeable improvement in emotional expressivity, particularly for high-intensity emotions such

as anger and sadness. Their ability to generate distinct movements aligned with the described emotional states became more reliable, suggesting that explicit bodily behavior cues helped the models create more meaningful animations. *T2M-GPT*, which relies on a transformer-based architecture, required even more detailed descriptions to produce consistent emotional variations. Prompts that included multiple bodily cues, such as "a person walks with slow, dragging steps, hunched shoulders, and head tilted downward", yielded better results than simpler formulations. Meanwhile, *Muse* continued to perform well with both simple and detailed prompts, reinforcing the hypothesis that its underlying model integrates emotional concepts more effectively than the others.

4.2 Implementation of the Unity-based Tool

4.2.1 Environment configuration

To effectively integrate these generative models into Unity, a careful management of the Python environment was essential. Virtual environments are a key tool for managing Python projects that require specific libraries or versions of them, or even different versions of Python itself. The use of virtual environments avoids conflicts between the dependencies of different projects, ensuring that each project has its own isolated space with the versions needed for proper operation. So, It is often necessary to enable a dedicated virtual environment for each model to ensure that all required libraries are properly installed and compatible.

The *Anaconda* [1] tool was chosen to manage the environments because it provides a very powerful and user-friendly interface for creating, managing, and deploying Python virtual environments. *Anaconda* simplifies package installation and setting up separate environments, all while supporting multiple versions of Python and complex libraries that might require special configuration. Using *Anaconda*, it is quite simple to set up and manage virtual environments for each generative model, which greatly ensures a smooth workflow tuned for the particular needs of each project.

4.2.2 Output conversion

One of the main challenges in effectively integrating text-to-motion models within Unity was accurately converting their generated outputs into compatible formats. None of the models analyzed were designed for direct integration with graphics engines or game engines such as Unity. The outputs generated by the models

[1] `https://www.anaconda.com/`

consist of a description of human joint positions (XYZ coordinates) in numpy array format (***.npy***), as well as a video file (***.mp4***) showing a stick figure visualization of the result. However, the availability of joint positions alone introduces a significant challenge: to convert this data into a format that can be used by 3D software, such as BVH or FBX, the gap regarding joint rotation information must be bridged. Inaccurate conversion could introduce physical or visual inconsistencies, with undesirable effects on the animated characters.

A first alternative for converting the output is offered by a feature in some models, which allows animations to be generated by exploiting the SMPL [64] model. SMPL (Skinned Multi-Person Linear) model is a widely used statistical body model that can generate realistic human meshes controlled by pose and shape parameters. However, SMPL does not directly accept raw joint coordinates as input. Instead, it requires a specific set of parameters describing the body's pose (i.e., joint rotations) and shape (i.e., body proportions). To bridge this gap, the SMPLify optimization method is typically employed.

SMPLify is an iterative optimization process that aims to recover the pose and shape parameters of the SMPL model such that the joints of the generated mesh align as closely as possible with the target joint positions. Given a set of observed 3D joints, SMPLify minimizes a cost function that penalizes discrepancies between the SMPL-predicted joint locations and the provided target positions.

This optimization generally includes several terms: a data term that measures the Euclidean distance between the predicted and observed joints, and regularization terms that ensure the resulting pose remains within plausible human limits. The optimization is performed frame-by-frame in the case of an animation, which allows the generation of a temporally coherent sequence of SMPL meshes corresponding to the original motion.

Once the optimization converges, the resulting parameters can be passed to the SMPL model, which outputs a detailed 3D mesh of the human body in the specified pose (see Fig.4.1). Repeating this process over time enables the creation of a smooth and realistic animated mesh that mirrors the original joint-based motion.

This method is particularly useful when the input data consists of only sparse joint positions, as SMPLify allows the reconstruction of full-body geometry and articulation. Furthermore, it is actually the same used by some text-to-motion models to visualize their results, making it a natural choice for data conversion.

This approach produces a more complete file than just joint positions, providing for each frame:

- An **OBJ** file, representing the body mesh for each frame.
- A detailed description of the motion parameters, including:
    1. **motion**, global motion.

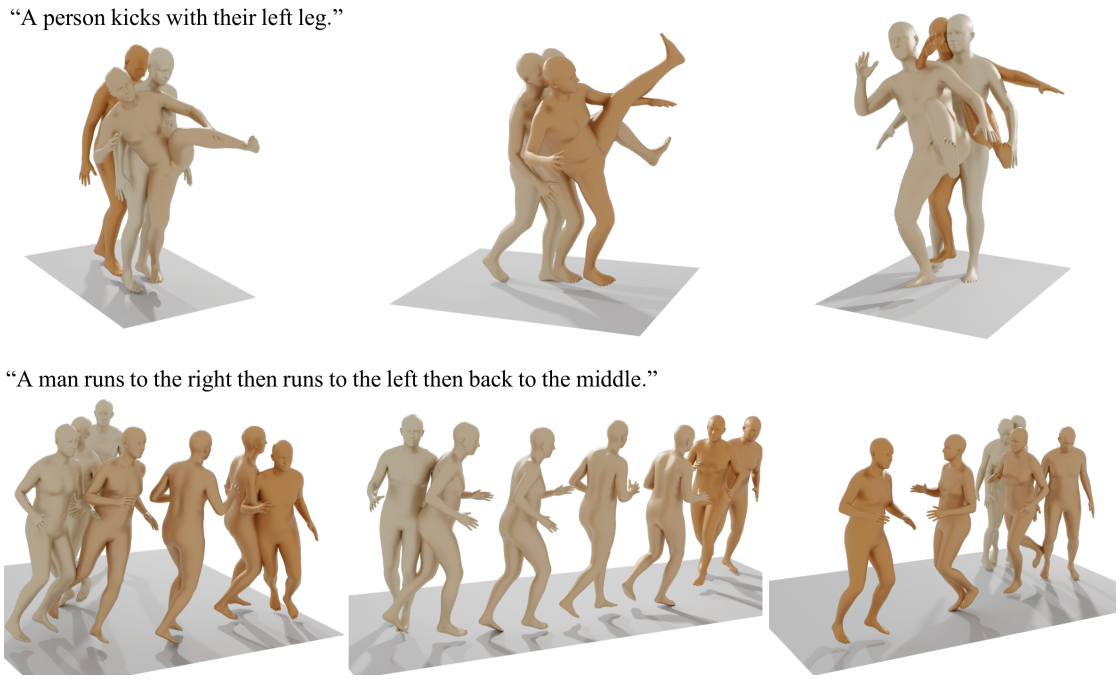


Figure 4.1: SMPL mesh for each frame of the animations [27]

2. **thetas**, joint rotations expressed in 6D format.
3. **root translation**, the translation of the root of the body in space.
4. **faces**, a list of the faces of the SMPL mesh.
5. **vertices**, the positions of the vertices of the SMPL mesh for each frame.
6. **text**, the textual prompt that generated the animation.
7. **length**, the total number of frames of the animation.

This algorithm needs GPU resources and takes a relatively long conversion time. Once the SMPL parameters have been calculated, an SMPL to FBX converter[2] can be used to transform the data into an FBX format, compatible with Unity. This method provides a detailed and physically accurate representation, providing good visual quality and consistency in animation.

Another approach for this conversion is to go through the **BVH** (Biovision Hierarchy) format, commonly used in motion capture. Since the BVH format requires both positions and rotations of the joints relative to their parent, it is necessary to compute these rotations from the positions alone. For this purpose, an

[2] `https://github.com/softcat477/SMPL-to-FBX`

IK (Inverse Kinematics) body solver[3] can be used, based on the observation that if the directions to which the joints point correspond to the directions of the vectors between the target joints, then the resulting pose will be aligned with the target pose. The solver's iterative process involves rotating each joint so that the vectors between the joint and its children coincide with those computed from the target poses. Once the BVH file has been generated, it can be exported as a Unity-compatible **FBX** file through Blender scripts. However, this method has some limitations, particularly for animations that include significant root rotations (e.g., rotations close to 360 degrees). In such cases, bugs, such as unnatural behavior of bones, can occur, compromising the quality of the animation. These issues can be mitigated by increasing the number of iterations of the IK solver algorithm, at the cost of longer generation times. Despite this, with a sufficient number of iterations, this method becomes accurate enough for this thesis's purposes, and it is the approach chosen for further development.

4.2.3 Integrating Python with Unity

To utilize the motion generation models mentioned earlier, it is necessary to configure the appropriate Python environment and execute commands via the shell. This process, entirely written in Python, must be integrated with Unity, which primarily uses C#. Therefore, a system was developed to enable sending commands directly from Unity's interface. The key steps in the process include:

1. Activating the Conda environment corresponding to the selected model.
2. Send the command to generate the animation.
3. Send the command to convert the animation into *FBX* format.

To establish communication between Unity and Python, two main approaches were tested:

- Using C# processes: launching terminal sessions and sending commands via StreamWriter.
- ZeroMQ [4]: a library enabling asynchronous and bidirectional communication between Unity and Python.

The final choice fell on ZeroMQ due to its flexibility and scalability, as it simplifies implementation for potential future extensions. Additionally, its asynchronous

[3] `https://github.com/sigal-raab/Motion`

[4] `https://zeromq.org/`

nature enhances performance by reducing bottlenecks and allows the use of the Unity editor even during generation.

However, the performance in terms of time of the two approaches does not show significant differences. This is because the Python server, even when using ZeroMQ, launches external processes via the subprocess library, introducing a similar overhead to that of direct execution from C#. Nonetheless, this overhead has minimal impact on animation generation, as the duration depends almost entirely on model generation and conversion.

Looking ahead, the goal is to import Python scripts directly as modules, allowing Unity to call Python functions without relying on the shell. This solution would eliminate the overhead associated with launching new processes.

Python-Unity Messaging System with ZeroMQ

ZeroMQ is an open-source messaging library that supports multiple messaging patterns, including REQ / REP (Request / Reply) and PUB / SUB (Publish / Subscribe). The REQ/REP pattern is straightforward: a client sends a request to a server, and the server replies. This enables bidirectional communication where the client retains control over the interaction. On the contrary, the PUB/SUB pattern allows a server to broadcast messages to all subscribed clients, creating a unidirectional communication flow where clients passively receive data.

For this implementation, the REQ/REP pattern was chosen to enable a structured exchange of commands and results between Unity and the Python server.

ZeroMQ supports exchanging any serializable data, from primitives to complex objects like images. In this project, communication involves *JSON*-formatted strings due to their simplicity and ease of manipulation.

Python Server Setup. The Python server initializes by importing the zmq library and creating a context. Within this context, a REP socket is instantiated and bound to a specific port (e.g., 5554) to listen for incoming messages from Unity. Although ZeroMQ supports various transport protocols, *TCP* is commonly used due to its reliability. The server runs in an infinite loop, waiting for client messages. Upon receiving a message, it processes the data and sends a reply back to Unity.

```
import zmq
context = zmq.Context()
socket = context.socket(zmq.REP)
socket.bind("tcp://*:5554")

while True:
    try:
        message = socket.recv_json()
```

```
9
10          prompt = message.get(...)
11          model = message.get(...)
12          [...]
13          result = execute_model(model, prompt, ...)
14          socket.send_string(result)
15
16      except zmq.ZMQError as e:
17          logging.error(f"ZMQ Error: {str(e)}")
18      except Exception as e:
19          logging.error(f"Unexpected error: {str(e)}")
```

Unity Client Setup. On the Unity side, a *RequestSocket* is used to establish communication with the Python server. Before initiating communication, the Python server is launched via a Unity script using the *System.Diagnostics.Process* class. Once the server is running, the Unity client establishes a connection and sends a *JSON*-formatted message containing the generation parameters. The client sends the *JSON* message to the Python server, waits for the response,and processes the returned result.

```
1  AsyncIO.ForceDotNet.Force();
2  client = new RequestSocket("tcp://localhost:5554");
3  try
4  {
5      var messageObj = new JsonMessage
6      {
7          prompt = promptText,
8          model = selectedModel,
9          [...]
10     };
11
12     var message = JsonUtility.ToJson(messageObj);
13     client.SendFrame(message);
14
15     if (client.TryReceiveFrameString(TimeSpan.FromSeconds(1), out
       string response))
16     {
17         processResponse = response;
18         break;
19     }
20
21 }
22 catch (Exception ex)
23 {
24     Debug.LogError("Exception: " + ex.Message);
```

```
25 }
26 finally
27 {
28     if (client != null)
29     {
30         client.Close();
31         ((IDisposable)client).Dispose();
32         NetMQConfig.Cleanup();
33     }
34     TerminatePythonServer();
35 }
```

4.2.4 Unity Editor

The most user-friendly way to allow users to generate animations directly within Unity is to use custom windows in the editor. Editor scripts in Unity are special scripts that extend the functionality of the game engine editor, allowing developers to create custom user interfaces, add advanced editing tools, and automate operations within the editor itself.

Unlike ***MonoBehaviours***, which are designed to be used during game execution and are attached to ***GameObjects***, editor scripts run exclusively in the Unity editor. ***MonoBehaviours*** allow code to be written that responds to game lifecycle events, while editor scripts focus on operations related to creating and modifying the scene and assets within the editor before the game is executed.

The main advantages of script editors lie in the ability to customize Unity's user interface and automate complex operations that would otherwise require multiple manual steps. Custom tools, such as windows, buttons, and context menus, can simplify interaction with the engine, improving efficiency during the development process.

This tool allows users to input parameters, trigger the generation process, and automatically apply the resulting animations to character rigs.

Interface Design

The custom Editor Window, illustrated in Fig. 4.2, consists of several key components designed to streamline the animation generation workflow:

1. **Text Prompt**: this field (Fig.4.2-1) allows users to enter the textual description of the desired motion, which is then sent to the selected generative model.

2. **IK Iterations**: this numerical input field (Fig.4.2-2) specifies the number of iterations for the inverse kinematics (IK) algorithm used during the conversion process. Higher values improve accuracy but increase processing time.

3. **Model**: dropdown menu (Fig.4.2-3)for selecting the desired generative model. The available options are displayed in a dropdown list, allowing users to switch between models with ease.

4. **Animation Duration**: when supported by the chosen model, this slider (Fig.4.2-4) controls the duration of the generated animation, enabling fine-tuning of motion length.

5. **Generate/Stop**: this button (Fig.4.2-5) starts or stops the animation generation and conversion process. When clicked, the system triggers the model execution, processes the generated data, and converts it into *FBX* format.

6. **Working Paths**: these fields (Fig.4.2-6) display the current working directories for the Python path and the server path. These paths are essential for the tool to locate the necessary scripts and resources.

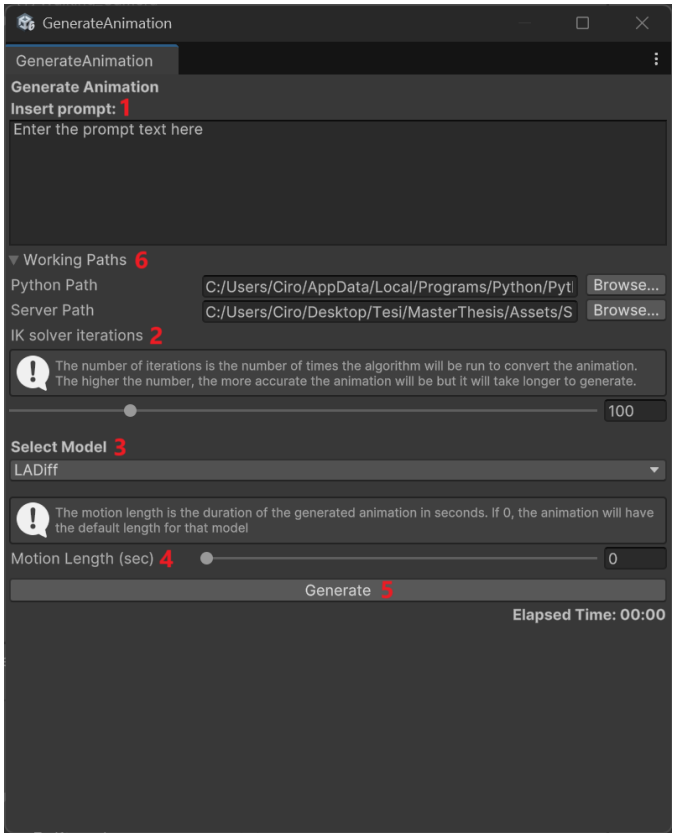


Figure 4.2: A screenshot of the *GenerateAnimation* script window.

4.2.5 Process overview

To systematically generate the animation video dataset, this Unity automated tool, given a motion model (m) and a text prompt encapsulating both the action (a) and the emotion (e), produces a recorded video clip under consistent rendering conditions.

In simple terms, animations were generated through an integrated pipeline involving two main environments: a **Python environment** for inference and animation data preparation, and a **Unity environment** for rendering and recording (see Fig.4.3).

The Unity client collects user input from the editor window (user interface, **UI**), which is then sent via *ZeroMQ* to the Python server. The server processes this input and retrieves execution parameters from a *JSON* configuration file, which includes paths to the generative models and the necessary commands to run them. By using a *JSON*-based configuration file, the process remains fully customizable; developers can easily modify the file to change model paths, adjust execution parameters, or add new models without needing to alter the code.

The server then executes the relevant commands based on the selected model, applies the IK solver, runs a Blender script for FBX conversion, and finally saves the resulting file directly in Unity's Resources folder. After the generation and conversion, the Unity client applies a Humanoid rig to the generated FBX file, extracts the animation clip, and deletes unnecessary files. The animation is then assigned to the Unity Animation System.

Finally, Unity Recorder was used to capture 1280x720 **MP4** video clips of each animation.

To avoid potential bias or confounding variables related to gender, age, or ethnicity—factors beyond the scope of this preliminary study—a neutral, mannequin-like character for all animations was employed. Since the chosen generative models do not currently animate hands, fingers, or faces, using a simplified mannequin prevents incomplete or distracting content that might otherwise distort participants' emotional perception [16, 57]. This approach is especially well suited to this exploratory research, as it provides a controlled assessment of how effectively the generated movements convey emotion solely through body language.

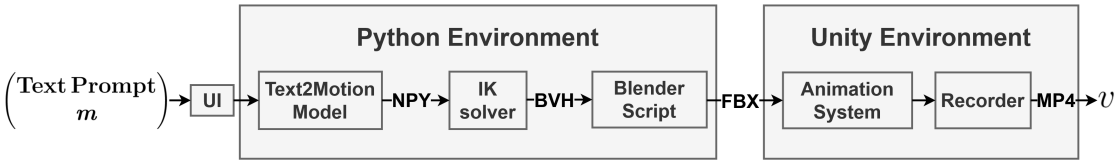


Figure 4.3: Process overview

Chapter 5

Experiments

This chapter describes the experiments conducted to evaluate the quality and effectiveness of the generated animations. The evaluation process includes two distinct experimental setups, each designed to assess different aspects of the animations: emotional expressiveness and alignment with textual prompts. The subsequent sections provide the detailed voting system and metrics definitions for each experiment.

5.1 Voting System

The user evaluation phase was conducted through an online survey managed by a custom web-based system responsible for distributing the animation video clips and collecting responses. The survey link was publicly shared—primarily among students of Politecnico—on a voluntary basis. Before participating, individuals provided basic demographic information (age, gender, nationality), which was used exclusively for aggregated analysis without storing any personally identifying data.

The system assigns each participant a personalized, randomized queue with all the animation clips to be evaluated. This randomized order was intended to prevent similar animations (e.g. those depicting sadness) from being grouped together, which could lead to a bias in the rating. The participants were then presented with the videos one after the other.

5.2 Metrics

5.2.1 Promp-coherence experiment

For each video v, they were asked to rate how well the prompt matched the animation (rated affinity, r_i) on a Likert scale from 1 (Low Affinity) to 7 (High

Affinity), generating a selection $s_i = (v_i, r_i)$. Figure 5.1 shows the survey's voting interface.

Figure 5.1: First experiment interface

Considering a desired S as a subset of the total selections, filtered by any combination of m, e^{gen} and a, **Affinity Score** R_s is calculated.

Affinity Score R_s is defined as the mean value of the rated affinity r_i across all selections $s_i \in S$. This value is used to assess how well the generated animation clips align with the given textual prompt. Higher values indicate a better correspondence between the generated animation and the described action. Collected values are represented on a seven-point Likert scale ranging from *low* affinity (one point) to *high* affinity (seven points) with *neutral* in the middle (four points), which are then mapped to a scale from zero to one, thus $R_s \in [0, 1]$.

5.2.2 Emotion perception experiment

Participants were asked to watch each animation exactly once and then select which of the six basic emotions (*happiness, sadness, anger, fear, surprise, disgust*) they believed it conveyed. They also rated the difficulty of making that decision on a seven-point Likert scale, ranging from *easy* (1) to *hard* (7). Thus, each selection (s_i) was recorded as (v_i, e_i^s, d_i) where v_i indicates the depicted video, e_i^s is the emotion selected by the participant, and d_i is the difficulty rating.

Figure 5.2 illustrates the survey's voting interface. By combining emotion selection and difficulty assessment, it has been gathered not only recognition accuracy—how well participants identified the intended emotion—but also insights into each participant's subjective confidence or uncertainty in making that choice.

This dual-metric approach allowed for a more nuanced evaluation of how effectively each generative model conveyed emotional expressivity in its animations.

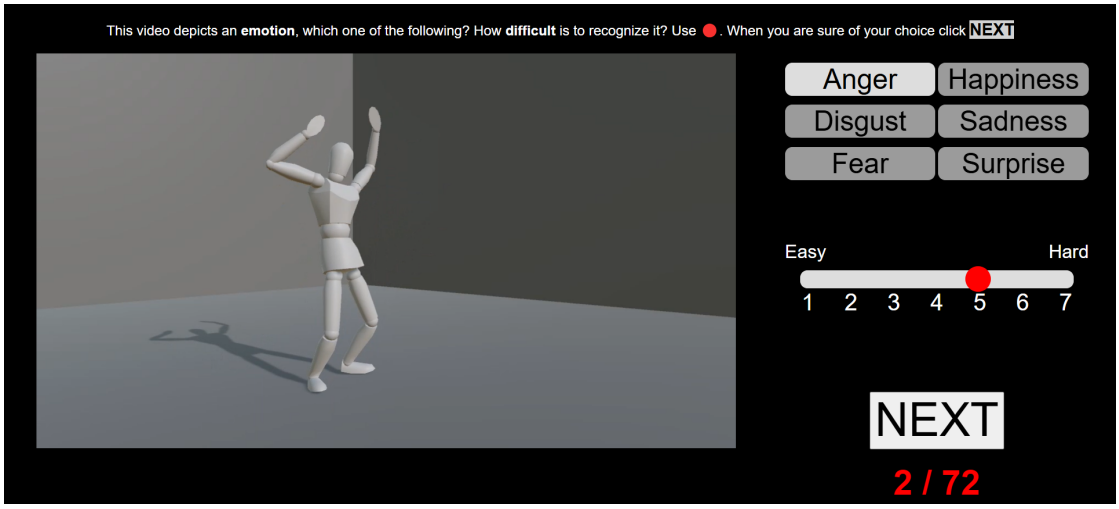


Figure 5.2: Second experiment interface

To evaluate performance at different levels of granularity, different subsets of all user selections were extracted to analyze specific factors (e.g., a single emotion, a model, an action, or any combination thereof). In this way, for example, metrics can be calculated only for the *anger* animations generated by model A or for the happiness on walking animations generated by all models.

Let S be an arbitrary subset of the user selections obtained by filtering for any combination of model (m), action (a), and intended emotion (e^{gen}).

The **accuracy** A_S defined as the number of correctly recognized emotions divided by the total number of selections in S and the **difficulty** D_S as the normalized mean of the difficulty d_i of all selections $s_i \in S$, i.e. $D_S = (\mu - 1)/6$ where $\mu = \frac{1}{n}\sum_{i=1}^{n} d_i$. Therefore, an accuracy closer to 1 corresponds to higher believability, while an accuracy closer to 0 corresponds to lower believability. Conversely, higher difficulty values mean greater uncertainty in identifying the emotion depicted. Intuitively, a negative correlation between A_S and D_S is expected, i.e. lower believability is associated with higher difficulty.

Chapter 6

Results

6.1 Emotion perception experiment

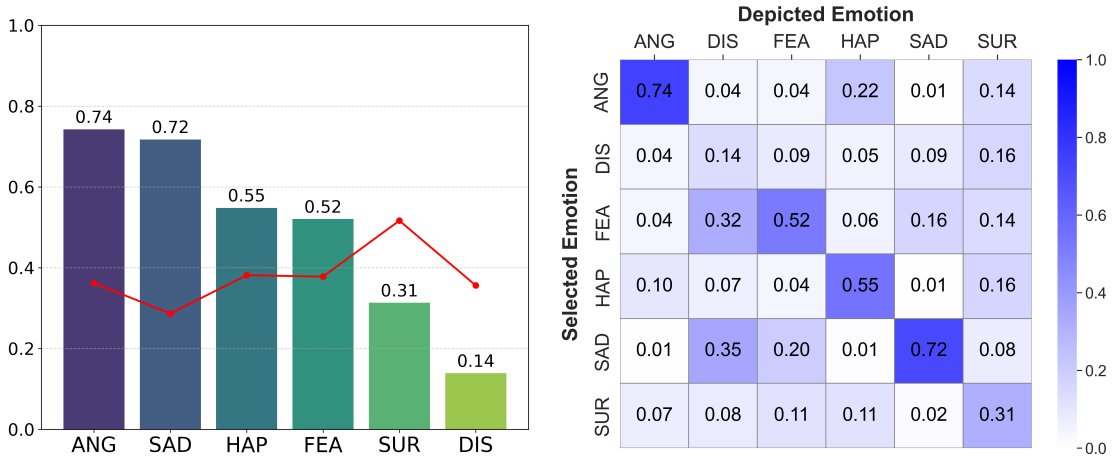
The primary aim of this study was to explore the believability and human recognizability of emotionally expressive animations generated by different text-to-motion AI models. Specifically, recognition accuracy across various emotions and actions was analyzed to gain preliminary insights into their potential to convey realistic emotional states. Given the exploratory nature of this research, it is crucial to contextualize the results within existing body language literature. By qualitatively comparing this study's findings with previous studies focused on human recognition of emotions from real human motion [65, 66, 67, 68], it can better understand whether the limitations observed in this study are specific to the virtual and AI-generated context or if they reflect broader challenges inherent in recognizing certain emotions through body language alone.

Participants

The survey resulted in a total of 39 participants with 33 who fully completed the survey, the others' selections were discarded. Participants were nine females and 24 males, of which 31 of them are Italian with a mean age of 27 ± 7 years old. Overall, the data collection resulted in 2376 selections, with a mean selection time of 23.75 seconds for a single video.

6.1.1 Emotions analysis

Considering overall metrics subdivied by emotions 6.1a, *anger* and *sadness* achieved the highest recognition accuracies (0.74 and 0.72, respectively). The higher accuracy for anger can be attributed to the presence of distinct body movements



(a) A_S (bars) and D_S (red line) **(b)** Confusion matrix

Figure 6.1: Aggregate results grouped by *emotion*.

typically associated with intense emotions—such as abrupt gestures, tense muscular postures, and forward-leaning stances—that are universally identifiable and consistently documented across multiple body language studies [68, 65]. Similarly, *sadness* benefits from highly recognizable bodily expressions such as slumped shoulders, lowered heads, and minimal limb movements [67, 66], providing clear visual cues even in the absence of facial details [68]. Conversely, *disgust* and *surprise* recorded significantly lower accuracy rates (0.14 and 0.31, respectively). The particularly low accuracy for *disgust* highlights its reliance on subtle facial expressions, including movements around the mouth and nose [57], typically difficult to convey effectively using body motion alone [66, 65]. *Surprise*, despite generally involving noticeable bodily cues such as sudden backward movements or raised limbs, suffers from ambiguity in body posture interpretation without complementary facial or hand cues, leading to increased confusion and lower recognition accuracy [68]. Additionally, considering that usually surprise emotion is expressed toward a sudden occurring environmental event[57], the lack of context may have added more selection uncertainty [67, 66]. *Happiness* achieved moderate accuracy (0.55), reflecting the complexity in interpreting this emotion solely through body language. Happiness is often expressed through dynamic and energetic gestures such as bouncing, rhythmic limb movements, or open and expansive postures [67, 68]. However, these cues can closely resemble other high-arousal emotions, resulting in ambiguity and recognition challenges when isolated from facial expressions or contextual elements [68, 66]. Similarly, *fear* showed moderate recognition accuracy (0.52), indicating variability and complexity in bodily expression. Fear can manifest in multiple distinct motor behaviors—ranging from freezing and subtle defensive

postures to overtly escaping movements—making it inherently context-dependent and less universally recognizable without facial or environmental context cues. These findings underscore well-documented challenges in previous studies, where subtler and context-dependent emotions heavily rely on specific expressive details harder to isolate in body movements alone [68, 66].

To gain insights on the incorrect selection of emotions, it is useful to consider their confusion matrix 6.1b. *Disgust*, frequently confused with *sadness* (0.35) and *fear* (0.32), has consistently been recognized as difficult to identify through body motion alone, as it heavily relies on subtle, facially-centered cues—particularly around the mouth and nose—that are not effectively conveyed through posture or gross motor gestures [57, 66, 65]. Prior research has emphasized that, when facial and hand information is absent, observers tend to misclassify it with emotions exhibiting similarly contracted or protective postures—i.e. sadness or fear— [67]. Similarly, *surprise* was commonly confused with all of the others, aligning with past literature indicating that surprise is inherently ambiguous without complementary facial cues, as its bodily expression often overlaps with positive (e.g., happiness) or negative (e.g., disgust, fear) reactions due to sudden and expansive gestures [68]. *Fear*, while moderately well recognized (0.52), was most commonly misclassified as *sadness* (0.20), hinting to common perceptual cues. This confusion is consistent with prior findings showing that fear-related movements can vary widely—from freezing and crouching to defensive or escape behaviors—and that, in the absence of facial expressions or contextual stimuli, such motions may resemble the inward, subdued posture of sadness [67, 66]. Equivalently, *happiness* was misclassified as *anger* (0.22), revealing a perceptual overlap in their bodily expressions. Prior studies have shown that both emotions are conveyed through expansive, high-energy movements—such as arm-raising or vigorous gestures—which, in the absence of facial features, can appear visually similar [68, 67, 66]. This suggests that observers may rely on motion dynamics as a proxy for intensity or arousal, leading to systematic confusion between highly activated emotional states. Conversely, *anger* and *sadness* showed fewer confusions with other emotions, consistent with previous findings highlighting their universally recognized and distinct body cues, such as forward-leaning and tense postures for anger, and closed, slouched postures for sadness [68, 66, 65].

Across all emotions, difficulty ratings range from easy to neutral. This outcome hints that even if the actual selected emotion was incorrect, participants perceived the produced animations as fairly coherent and natural. Although this distribution implies a negative relationship between recognition correctness and perceived difficulty, a strong monotonic correlation does not materialize 6.1a (Spearman's rank correlation resulted in $\rho = -0.15$, $p < 0.001$, and a 95% confidence interval of $[-0.19, -0.11]$). Similar results are obtained considering correctness and emotion selection times (Spearman's rank correlation resulted in $\rho = -0.10$, $p < 0.001$,

and a 95% confidence interval of $[-0.14, -0.06]$). Contrarily, an interesting finding emerges when examining the relationship between selecting time and perceived difficulty. While accuracy shows no significant correlation, a moderate and statistically significant monotonic relationship is observed between voting time and difficulty ratings (Spearman's rank correlation resulted in $\rho = 0.44$, $p < 0.001$, and a 95% confidence interval of $[0.41, 0.47]$). This suggests that participants who perceived a stimulus as harder also took longer to cast their selection, reflecting greater cognitive effort or uncertainty in emotion attribution.

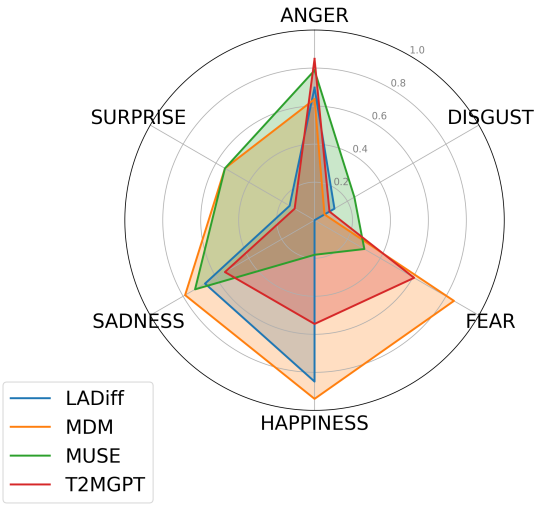
Varying accuracies observed across emotions emphasize the importance of distinctive and universal non-verbal cues in accurately conveying emotional states in VHs animation as well as real human motion. Emotions characterized by well-defined, cross-culturally consistent gestures and postures, such as anger and sadness, are more successfully communicated through body movements, whereas emotions that rely predominantly on subtle or facial-dependent details, such as disgust and surprise, present inherent limitations for effective recognition without complementary expressive channels.

6.1.2 Actions and Models analysis

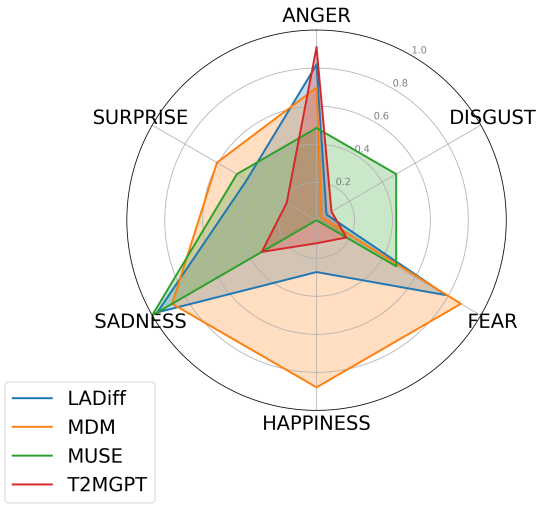
To better interpret the performance outcomes, it is useful to take a closer—though not exhaustive—look at how different generative models behave across various actions and emotional contexts. While the analysis that follows is only preliminary, it attempts to hypothesize potential explanations for model behavior by examining their outputs in relation to specific body movement cues and emotional expressions. These interpretations are speculative and should be considered as an initial step toward understanding the models' capabilities, informed by body language literature and the structural characteristics of the models involved.

Speaking

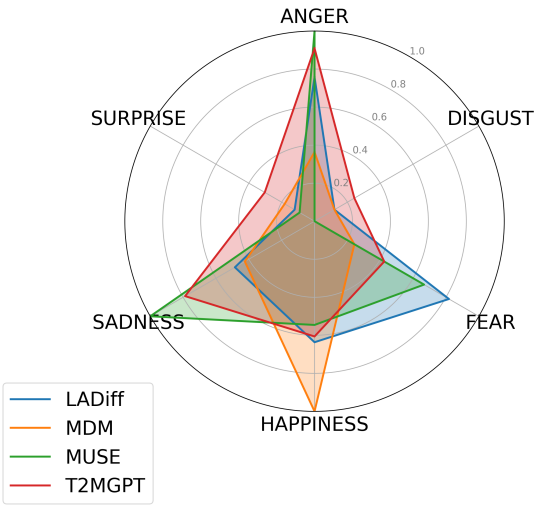
MDM demonstrates relatively higher accuracy across multiple emotions 6.2a, especially for *happiness*, *fear*, *sadness*. While these results might partly stem from the universally expressive nature of these emotions [67, 65], the model's transformer-based architecture, which is designed to manage the temporal and spatial irregularities of motion data, may also support slightly more coherent and believable emotional animation in speech scenarios. While this remains speculative in terms of expressive clarity, the model's use of geometric losses on joint locations and velocities [27] suggests a technical basis for improved motion continuity. Muse and LADiff follow, performing moderately well, in particular better than MDM on *anger*, potentially benefiting from possible learned emotional embeddings for Muse, or diffusion-based smoothing for LADiff [28], which may stabilize emotional



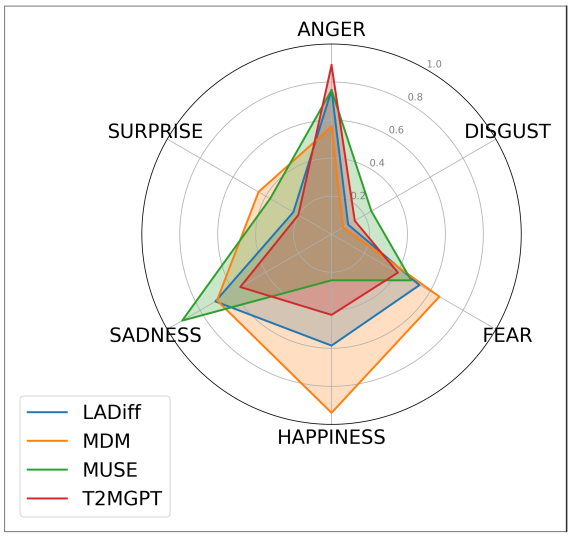
(a) *Speaking* accuracy.



(b) *Standing* accuracy.



(c) *Walking* accuracy.



(d) Overall accuracy.

Figure 6.2: Aggregated accuracies for models and emotions. Subfigures (a), (b) and (c) by actions, subfigure (d) overall.

portrayals in specific Speaking poses. Interestingly, LADiff's accuracy regarding *fear* is close to zero due to its consistent confusion with *sadness* (ratio of 0.76), probably due to inherent animation characteristics leading to consistent ambiguity as seen in overall findings 6.1b. This confusion likely stems from the model's latent space denoising process, during which emotional nuances may be overly smoothed or collapsed, particularly for emotions with overlapping low-energy postures like fear and sadness. In contrast, this confusion is less evident in MDM,

which does not rely on a latent-space diffusion mechanism and instead generates motion sequences directly in pose space, preserving more distinct structural features between similar low-activation emotions. On the other hand, the non-diffusion architecture of T2MGPT, relying on tokenized motion representation [45], appears less consistent, possibly due to the discretization of motion into symbolic units that can limit fluidity and nuance. While this process enables structured prediction and scalability, it may constrain the model's ability to represent continuous and context-sensitive emotional cues, especially in subtle or complex gestures. On the other hand, this architectural characteristic might interestingly enhance T2MGPT at portraying *anger*, in which the required strong and abrupt movements might be naively portrayed by the intrinsic limited motion fluidity.

The portrayals of *surprise* and *disgust* remain particularly challenging across all models, which is consistent with broader findings in body language literature [68]. While the overall performance remains limited, MDM and Muse appear to handle basic emotions slightly better.

Standing

Again, MDM shows slightly higher accuracies, particularly for *fear*, *happiness*, and *sadness*. This might reflect the model's ability to capture subtle posture changes even in limited-motion contexts, possibly benefiting from its transformer-based structure [27]. Standing poses inherently offer fewer dynamic cues, making emotion conveyance more reliant on posture and subtle weight shifts [67, 65], areas where MDM's design might offer an advantage. LADiff's performance only slightly lags behind MDM overall due to a noticeable dip in expressing happiness, which seems to stem from specific characteristics of the generated animations—mainly confused with *anger* with a 0.52 rate, consistent to overall results 6.1b. Muse shows modest improvements in conveying ambiguous emotions like *surprise* and *disgust*. This may be due, similarly to Speaking action, to mechanisms that promote stylistic coherence or learned patterns from expressive training examples. These results remain tentative and should be interpreted cautiously—in particular regarding Muse, being undocumented—, especially as suprise and disgust emotions continue to challenge recognition even under optimal modeling conditions [68]. Lastly, T2MGPT is outperformed by other models except for *anger*, where it excels among others, possibly reflecting its capacity to tokenize the tense posture shifts associated with the emotion. This difference remains speculative and warrants further verification but highlights how even minimal shifts in stance may be expressed more distinctly by some architectures than others.

Walking

Results show a heterogeneous landscape, with each model highlighting a different emotional strength. Muse shows relative proficiency in conveying *anger* and *sadness*, possibly owing to naturally embedded emotional comprehension, which may help capture the slow, heavy gait characteristic of sadness-related motion. In the case of *anger*, the model may benefit from its ability to correctly reproduce increased joint rigidity and greater arm swing amplitude, with sharper elbow and shoulder angles—features often associated with high-arousal, confrontational body language [68]. As well, T2MGPT captures *anger* effectively, which might be due again to its token-based generative mechanism, enabling sharp and intense motion bursts that align with the high-activation, aggressive gait typically associated with anger in body language research [66]. Conversely, LADiff appears somewhat better at portraying *fear*, likely benefiting from its latent diffusion process that emphasizes smooth temporal coherence while still allowing for subtle retreating or defensive kinematic cues typical of fearful walking [68]. Meanwhile, MDM appears to excel at conveying *happiness*, presumably due to its transformer-based architecture's ability to capture the vibrant, upper-body gestures often associated with this upbeat emotion. Nonetheless, *surprise* and *disgust* continue to yield poor results across all models, highlighting the persistent difficulty of conveying these emotions through body movement alone [68, 65].

Overall

Overall, MDM and Muse perform slightly better than the others, though all models show room for improvement 6.2d. MDM overall succeeds in conveying nuanced emotional cues across different actions, possibly due to its transformer-based architecture and use of geometric loss functions, which support more structured motion sequences. Muse shows a degree of emotional expressivity, perhaps due to training on motion-emotion semantics, which helps it capture some recognizable cues, especially in high-energy emotions like anger. LADiff and T2MGPT show limitations overall. LADiff struggles particularly with emotions like fear and happiness, sometimes confusing them probably due to its latent diffusion approach, which may blur subtle emotional differences during denoising. T2MGPT, utilizing tokenization-based generation, struggles with fluidity and subtlety in most emotions but interestingly excels at anger, where abrupt and sharp movements align well with tokenization strengths.

In conclusion, while all models exhibit considerable limitations—especially in conveying complex or subtle emotions like surprise and disgust—MDM and Muse offer comparatively better performance. Their relative advantages suggest that certain architectural choices and training approaches may provide a more promising direction for improving emotional motion generation, though much work remains

to be done.

6.2 Prompt-coherence Experiment

This experiment was conducted prior to the emotion perception experiment to validate the prompts used. However, this section discusses the results retrospectively, incorporating insights gained from the outcomes of the subsequent emotion recognition experiment to provide explanatory context.

The user evaluation phase registered 21 participants, 5 females, and 16 males, all of whom are Italian. Most participants are $20 \div 30$ years old, with only one of them over 50. Overall, the data collection resulted in 1512 selections, with a mean selection time of 34.27 seconds for a single video.

6.2.1 Overall Analysis

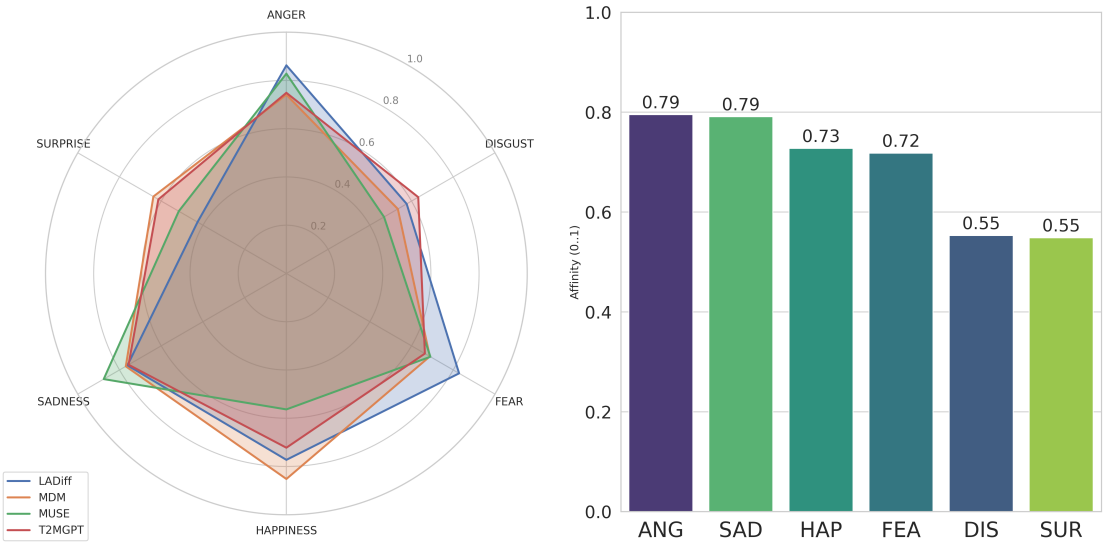


Figure 6.3: Overall *affinity* across *models* and *emotions*.

Considering the affinity score grouped by *models* and *emotions* in Fig. 6.3, the highest-ranked emotions remain the same: Anger and Sadness. This indicates that the animations for these emotions were clearly recognizable to users and consistent with the initial prompt. It also suggests a strong correspondence between the bodily behaviors described in the prompt and the conveyed emotion.

In contrast, the other ones exhibit a more noticeable divergence between affinity and accuracy. Although users rated the alignment between the prompt and the

generated animation relatively high ($R_s > 0.55$), they had greater difficulty correctly identifying these emotions compared to Anger and Sadness. As illustrated in the confusion matrix (Fig. 6.1b), this is likely due to the frequent misclassification of these emotions, which share similar bodily expressions with others: Happiness is often confused with Anger, Fear with Sadness, and Disgust with both Fear and Sadness. Thus, while the animation may closely match the prompt, it may still be misinterpreted in the emotion recognition experiment.

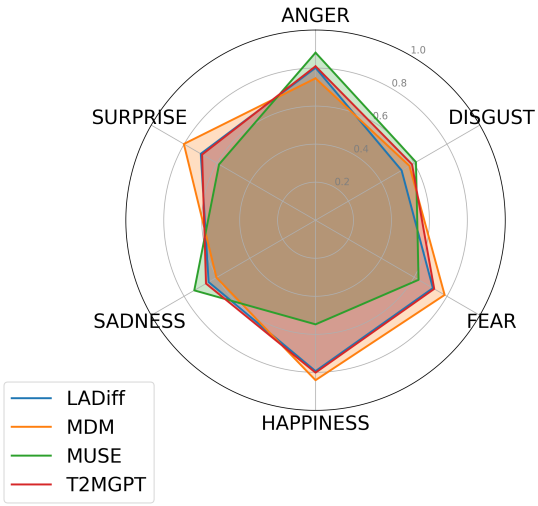
Regarding Surprise, as noted in the previous experiment, this emotion remains particularly challenging for current AI models to reproduce due to its unique characteristics. Additionally, research on body language indicates that Surprise is not always clearly identifiable through gestures alone [69, 70]. This ambiguity could explain why the affinity scores were relatively acceptable. For instance, the T2M-GPT prompt for Surprise during a speaking action is described as *"A person is talking with gestures and then suddenly lifts their arms"*. Although the mannequin may not have lifted its arms abruptly as intended, it still performed the described movement, leading participants to assign ratings above the neutral level (four out of seven). Nevertheless, the underlying emotional intent of Surprise remained ambiguous to users.

As regards the models, MDM and LADiff best adhere to the given prompt. Surprisingly, Muse performed the worst in this aspect, despite ranking second in emotion perception. This discrepancy is likely due to the nature of the prompts used in the emotion evaluation, which were more general, such as *"A person is sad"*. In this case, participants may have judged based on simple cues—such as a character with its head down—which, when compared to distinct emotional categories, strongly suggests that sadness is the right choice. However, when evaluating prompt affinity, they might have questioned whether the model's output truly conveys sadness to its full extent (i.e., if it is deeply expressive or just minimally suggestive).

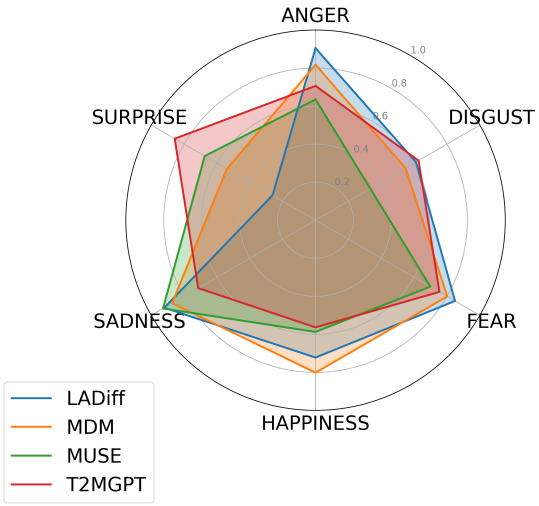
6.2.2 Action Analysis

Speaking

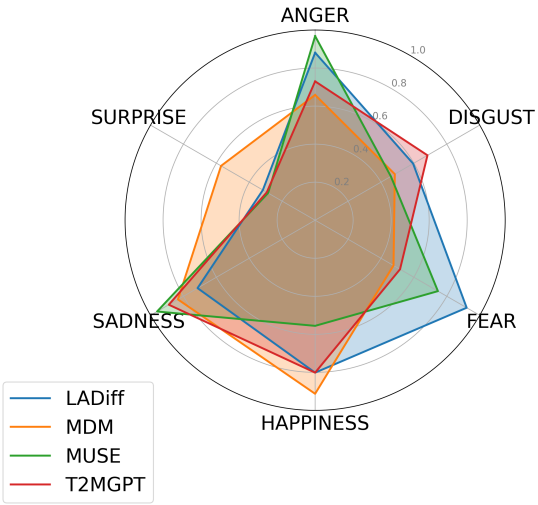
Considering the *Speaking* action (Fig. 6.4a), all models show a generally consistent performance, with values clustering around the 0.6-0.8 range for most emotions. There is no single model that clearly outperforms in all emotions. MUSE is the best for Anger, Sadness, and Disgust, while MDM leads in the other three (Surprise, Happines, and Fear). Observing the accuracy for T2M-GPT and MUSE in Fig. 6.2a (on the right), it is evident that Happiness is not well recognized. However, it achieves good results in prompt affinity, likely due to its bodily behaviors being similar to those of Anger. The same for Disgust misclassified with Sadness or Fear.



(a) *Speaking* action



(b) *Standing* action



(c) *Walking*

Figure 6.4: *Actions'* affinity

Standing

Unlike the *Speaking* action, where models were relatively balanced, there is greater variation in how models interpret emotional cues in the *Standing* pose (Fig. 6.4b). LADiff excels in Anger and Fear, achieving the highest affinity. But struggles in Surprise. MDM is more balanced. MUSE shows the strongest affinity for Fear and Anger. T2M-GPT performs unexpectedly well in terms of affinity for Surprise, achieving the highest score despite having the lowest emotion accuracy. This is likely due to the lack of sudden movements typically associated with Surprise,

making the emotion less distinguishable even if the generated motion aligns with the prompt. The description itself—*"A person stands still then suddenly makes a step backward toward an unexpected stimulus"*—can result in a deceptively high-affinity score, as even a simple backward step may be sufficient to fulfill the prompt.

Walking

The variation across models is more pronounced than in *Standing* and *Speaking* actions, suggesting that emotional cues are harder to capture while walking (Fig. 6.4c). LADiff excels in Fear and Anger, MDM is best in Happiness as expected, MUSE is the strongest model for Anger and Sadness, and T2M-GPT in Disgust.

6.2.3 Limitations and Future Works

Due to the preliminary nature of this study, some important limitations, which are discussed in the following, should be noted.

Firstly, the used prompt engineering iterative method for animation clips generation, although being systematically conducted and evaluated with a pre-screen user study, may still have introduced an evaluation bias. Text-to-motion bases itself on text input which needs to be carefully crafted to achieve desired motion results. This is a well-known complex task for AI text-to-image models [71] but still unexplored for text-to-motion ones. Future research should focus on this aspect, evaluating different prompting strategies applied to the broad text-to-motion task as well as the ETMG one.

Secondly, as mentioned previously, cues such as finger gestures, facial expressions, and general visual appearance heavily influence emotional perception. Furthermore, the gender-neutral appearance of the model made possible perception differences between male and female VHs unfeasible, thus limiting this analysis. Future research should explore diverse character models with explicit demographic attributes and possibly consider adding hands and facial animations to better understand how these variables influence the emotional believability of generated animations.

Thirdly, generative models were trained on data without explicit emotional state labeling, with the only notable exception of L3EM [29], which has no publicly available code by now. This constrains the models' ability to accurately interpret and create fine-grained emotional body language. Future research should focus on developing specific datasets for ETMG and training state-of-the-art models to directly encode emotional motion nuances—e.g., posture, velocity, acceleration, and amplitude—into neural network embeddings, with the aim of naturally reproducing them.

Fourthly, this exploratory study restricted the set of emotions explored and its application. Body language research has also investigated emotions beyond

Ekman's basic set, such as Shame and Pride [16], revealing distinctive associated cues. Moreover, only three daily actions were considered, significantly limiting the investigation within typical Virtual Human (VH) scenarios. Future research must broaden its exploration to include a wider range of emotions, behaviors, and complex interactions—particularly those involving objects or other VHs—in immersive XR environments. Crucially, the actual animations generated should be tested and evaluated directly in VR, where embodiment, presence, and spatial perception can significantly influence the interpretation of nonverbal behavior.

Finally, the vast majority of participants were Italian with ages spanning from 20 to 30 years old. Participants' evaluation could have been affected because of possible cultural differences, referring to nationality, and expectations regarding computer-generated animation videos, especially referring to age. Future research should try to diversify its participants' demography in order to reduce any potential bias.

Chapter 7

Conclusions

The advancements in AI-driven animation synthesis explored in this thesis demonstrate the potential of generative models, particularly diffusion-based architectures, in creating expressive and realistic motion sequences. Through a comprehensive evaluation of state-of-the-art models, it is evident that AI-generated animations have made significant strides in terms of quality and coherence.

The varying accuracies observed across emotions underscore the central role of distinctive and culturally consistent non-verbal cues in conveying emotional states, not only in human communication but also in AI-driven text-to-motion animation. The experiments results, derived from the generation of emotional full-body motions based solely on textual emotion prompts, reflect patterns consistent with findings from body language research. Emotions such as *anger* and *sadness*, which are associated with well-defined and universally recognizable postures and movement patterns, are more accurately conveyed by text-to-motion models. In contrast, emotions like *disgust*, *surprise*, and to a moderate extent *happiness* and *fear*, rely more heavily on subtle cues or facial expressions, making them inherently more difficult to represent through body motion alone. While these limitations highlight the current boundaries of text-to-motion systems, the observed alignment with established non-verbal communication literature suggests that this approach may hold strong potential. With further development, such as integrating multi-modal cues or refining body-part-specific expressiveness, and more in-depth investigation on models architecture regarding animation dynamic—considering emotion-related parameters such as posture, motions' speed, acceleration and amplitude—text-to-motion could become a valuable tool for generating believable emotional animations.

7.1 Limitations and Future Works

In addition to the challenges related to emotional expressiveness, current text-to-motion models also present technical limitations that need to be addressed to enhance their applicability in real-world scenarios. Challenges remain in ensuring fine-grained control, physics realism, and prompt adherence. Furthermore, future research should focus on:

- **Real-time applicability:** current generative text-to-motion models are generally not optimized for real-time animation, which limits their use in interactive applications and gaming. While initial efforts are emerging in this direction, real-time capabilities remain an underexplored and rapidly evolving research area.
- **Built-in solutions:** most generative models are not inherently designed for game development workflows, often requiring additional conversion steps that may be inefficient or suboptimal. Future research should prioritize aligning text-to-motion models with game engine requirements, similar to initiatives like Unity Muse, to improve overall workflow efficiency.
- **Human-Object Interaction (HOI) and Multi-Character Animations:** existing generative models predominantly focus on single-character scenarios and lack robust capabilities in generating coherent interactions involving multiple characters or precise manipulations of objects within the environment. Future work should emphasize developing models capable of understanding and producing context-aware animations that realistically depict interactions between multiple characters and their environment, including accurate representation of object usage, collaborative tasks, and complex social dynamics.

Ringraziamenti

Dopo anni di studio e impegno, finalmente questo lungo percorso è giunto al termine.

Vorrei pertanto ringraziare la mia famiglia per avermi permesso di studiare al Politecnico e avermi sostenuto in ogni scelta.

Ringrazio poi tutti i miei amici del poli che hanno reso lo studio più leggero e divertente, tra una pausa caffè e l'altra.

Ringrazio infine il prof Bottino, il prof Strada e Stefano per avermi dato la possibilità di poter lavorare su ciò che mi piace e per essere stati sempre disponibili ad aiutarmi.

Bibliography

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML]. URL: https://arxiv.org/abs/1406.2661 (cit. on p. 4).

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762 (cit. on p. 4).

[3] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2022. arXiv: 1312.6114 [stat.ML]. URL: https://arxiv.org/abs/1312.6114 (cit. on p. 5).

[4] EugenioTL. *Own work*. https://commons.wikimedia.org/w/index.php?curid=107231101. Wikimedia Commons; CC BY-SA 4.0. 2021 (cit. on p. 6).

[5] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. *Deep Unsupervised Learning using Nonequilibrium Thermodynamics*. 2015. arXiv: 1503.03585 [cs.LG]. URL: https://arxiv.org/abs/1503.03585 (cit. on p. 5).

[6] Ziyi Chang, George Alex Koulieris, and Hubert P. H. Shum. *On the Design Fundamentals of Diffusion Models: A Survey*. 2023. arXiv: 2306.04542 [cs.LG]. URL: https://arxiv.org/abs/2306.04542 (cit. on p. 5).

[7] Ruiqi Gao Arash Vahdat Karsten Kreis. *CVPR 2022 Tutorial. Denoising Diffusion-based Generative Modeling: Foundations and Applications*. https://cvpr2022-tutorial-diffusion-models.github.io/. 2022 (cit. on p. 7).

[8] Rajat Arora. *Role of virtual humans in healthcare simulations*. May 2021. URL: https://elearningindustry.com/virtual-humans-in-healthcare-simulations (cit. on p. 9).

[9] Kate Loveys, Michael Antoni, Liesje Donkin, Mark Sagar, and Elizabeth Broadbent. «Comparing the feasibility and acceptability of a virtual human, teletherapy, and an e-manual in delivering a stress management intervention to distressed adult women: pilot study». In: *JMIR formative research* 7 (2023), e42390 (cit. on p. 9).

[10] Tingting Liu, Zhen Liu, Minhua Ma, Tian Chen, Cuijuan Liu, and Yanjie Chai. «3D visual simulation of individual and crowd behavior in earthquake evacuation». In: *Simulation* 95.1 (2019), pp. 65–81 (cit. on p. 9).

[11] Dylan GM Schouten, Agnes A Deneka, Mariët Theune, Mark A Neerincx, and Anita HM Cremers. «An embodied conversational agent coach to support societal participation learning by low-literate users». In: *Universal access in the information society* 22.4 (2023), pp. 1215–1241 (cit. on p. 9).

[12] Therese Johansson and Mirjam Palosaari Eladhari. «Emotional Believability of Non-playable Game Characters-Animations of Anger, Sadness and Happiness». In: *International Conference on Interactive Digital Storytelling*. Springer. 2024, pp. 72–99 (cit. on p. 9).

[13] Stefano Calzolari, Francesco Strada, and Andrea Bottino. «The Quest for Believability: Exploring FACS Adaptations for Emotion Facial Expressions in Virtual Humans». In: *2024 IEEE Gaming, Entertainment, and Media Conference (GEM)*. IEEE. 2024, pp. 1–6 (cit. on p. 9).

[14] Katie Seaborn, Norihisa P Miyake, Peter Pennefather, and Mihoko Otake-Matsuura. «Voice in human–agent interaction: A survey». In: *ACM Computing Surveys (CSUR)* 54.4 (2021), pp. 1–43 (cit. on p. 9).

[15] Zachary Meyer, Nicoletta Adamo, and Bedrich Benes. «Bodily expression of emotions in animated agents». In: *Advances in Visual Computing: 16th International Symposium, ISVC 2021, Virtual Event, October 4-6, 2021, Proceedings, Part II*. Springer. 2021, pp. 475–487 (cit. on p. 10).

[16] Zachary Witkower and Jessica L. Tracy. «Bodily Communication of Emotion: Evidence for Extrafacial Behavioral Expressions and Available Coding Systems». In: *Emotion Review* 11.2 (2019), pp. 184–193. DOI: 10.1177/1754073917749880 (cit. on pp. 10, 23, 26, 36, 51).

[17] Hillel Aviezer, Yaacov Trope, and Alexander Todorov. «Body cues, not facial expressions, discriminate between intense positive and negative emotions». In: *Science* 338.6111 (2012), pp. 1225–1229 (cit. on p. 10).

[18] Pernilla Larsson. *Discerning Emotion Through Movement : A study of body language in portraying emotion in animation*. 2014 (cit. on p. 10).

[19] Nele Dael, Marcello Mortillaro, and Klaus R. Scherer. «The Body Action and Posture Coding System (BAP): Development and Reliability». In: *Journal of Nonverbal Behavior* 36.2 (June 2012), pp. 97–121. ISSN: 1573-3653. DOI: 10.1007/s10919-012-0130-0. URL: https://doi.org/10.1007/s10919-012-0130-0 (cit. on p. 10).

[20] R. von Laban and L. Ullmann. *The Mastery of Movement*. First publ. 1950 under title 'Mastery of movement on the stage'. Macdonald & Evans, 1971. ISBN: 9780712113571. URL: https://books.google.ch/books?id=-RYIAQAAMAAJ (cit. on p. 10).

[21] Antja Kennedy. «60. Laban based analysis and notation of body movement». In: *Volume 1*. Ed. by Cornelia Müller, Alan Cienki, Ellen Fricke, Silva Ladewig, David McNeill, and Sedinha Tessendorf. Berlin, Boston: De Gruyter Mouton, 2013, pp. 941–958. ISBN: 9783110261318. DOI: doi:10.1515/9783110261318.941. URL: https://doi.org/10.1515/9783110261318.941 (cit. on p. 10).

[22] Bänziger, T. & Scherer, and Kr. «Introducing the Geneva Multimodal Emotion Portrayal (Gemep) Corpus». In: *A Blueprint for Affective Computing: A Sourcebook and Manual*. Ed. by Klaus R. Scherer, Tanja Bänziger, and Etienne Roesch. Oxford University Press, 2010 (cit. on p. 10).

[23] Wikimedia Commons. *File:Laban Categories.svg — Wikimedia Commons, the free media repository*. [Online; accessed 13-March-2025]. 2024. URL: %5Curl%7Bhttps://commons.wikimedia.org/w/index.php?title=File:Laban_Categories.svg&oldid=869132394%7D (cit. on p. 12).

[24] Maxime Garcia, Rémi Ronfard, and Marie-Paule Cani. «Spatial motion doodles: Sketching animation in vr using hand gestures and laban motion analysis». In: *Proceedings of the 12th ACM SIGGRAPH Conference on Motion, Interaction and Games*. 2019, pp. 1–10 (cit. on p. 11).

[25] Diane Chi, Monica Costa, Liwei Zhao, and Norman Badler. «The EMOTE model for effort and shape». In: *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. 2000, pp. 173–182 (cit. on p. 11).

[26] Björn Hartmann, Maurizio Mancini, and Catherine Pelachaud. «Implementing expressive gesture synthesis for embodied conversational agents». In: *International Gesture Workshop*. Springer. 2005, pp. 188–199 (cit. on p. 11).

[27] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. «Human Motion Diffusion Model». In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=SJ1kSyO2jwu (cit. on pp. 13, 16, 24, 30, 43, 45).

[28] Alessio Sampieri, Alessio Palma, Indro Spinelli, and Fabio Galasso. *Length-Aware Motion Synthesis via Latent Diffusion*. 2024. arXiv: 2407.11532 [cs.CV]. URL: https://arxiv.org/abs/2407.11532 (cit. on pp. 13, 17, 24, 43).

[29] Tan Yu, Jingjing Wang, Jiawen Wang, Jiamin Luo, and Guodong Zhou. «Towards Emotion-enriched Text-to-Motion Generation via LLM-guided Limb-level Emotion Manipulating». In: *Proceedings of the 32nd ACM International Conference on Multimedia*. MM '24. ACM, Oct. 2024, pp. 612–621. DOI: 10.1145/3664647.3681487. URL: http://dx.doi.org/10.1145/3664647.3681487 (cit. on pp. 13, 21, 24, 50).

[30] Saeed Ghorbani, Ylva Ferstl, Daniel Holden, Nikolaus F. Troje, and Marc-André Carbonneau. «ZeroEGGS: Zero-shot Example-based Gesture Generation from Speech». In: *Computer Graphics Forum* 42.1 (2023), pp. 206–216. DOI: https://doi.org/10.1111/cgf.14734. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14734. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.14734 (cit. on pp. 13, 20).

[31] Lianying Yin, Yijun Wang, Tianyu He, Jinming Liu, Wei Zhao, Bohan Li, Xin Jin, and Jianxin Lin. *EMoG: Synthesizing Emotive Co-speech 3D Gesture with Diffusion Model*. 2023. arXiv: 2306.11496 [cs.CV]. URL: https://arxiv.org/abs/2306.11496 (cit. on pp. 13, 20).

[32] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. *MotionDiffuse: Text-Driven Human Motion Generation with Diffusion Model*. 2022. arXiv: 2208.15001 [cs.CV]. URL: https://arxiv.org/abs/2208.15001 (cit. on pp. 15, 24).

[33] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, Jingyi Yu, and Gang Yu. *Executing your Commands via Motion Diffusion in Latent Space*. 2023. arXiv: 2212.04048 [cs.CV]. URL: https://arxiv.org/abs/2212.04048 (cit. on p. 15).

[34] Chongyang Zhong, Lei Hu, Zihao Zhang, and Shihong Xia. *AttT2M: Text-Driven Human Motion Generation with Multi-Perspective Attention Mechanism*. 2023. arXiv: 2309.00796 [cs.CV]. URL: https://arxiv.org/abs/2309.00796 (cit. on p. 15).

[35] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. «ReMoDiffuse: Retrieval-Augmented Motion Diffusion Model». In: *arXiv preprint arXiv:2304.01116* (2023) (cit. on p. 15).

[36] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. «Guided Motion Diffusion for Controllable Human Motion Synthesis». In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2151–2162 (cit. on pp. 16, 24).

[37] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. *OmniControl: Control Any Joint at Any Time for Human Motion Generation*. 2024. arXiv: 2310.08580 [cs.CV]. URL: https://arxiv.org/abs/2310.08580 (cit. on pp. 17, 24).

[38] Wenyang Zhou et al. *EMDM: Efficient Motion Diffusion Model for Fast and High-Quality Motion Generation*. 2024. arXiv: 2312.02256 [cs.CV]. URL: https://arxiv.org/abs/2312.02256 (cit. on pp. 17, 24).

[39] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H. Bermano. *Human Motion Diffusion as a Generative Prior*. 2023. arXiv: 2303.01418 [cs.CV]. URL: https://arxiv.org/abs/2303.01418 (cit. on p. 17).

[40] Korrawe Karunratanakul, Konpat Preechakul, Emre Aksan, Thabo Beeler, Supasorn Suwajanakorn, and Siyu Tang. *Optimizing Diffusion Noise Can Serve As Universal Motion Priors*. 2024. arXiv: 2312.11994 [cs.CV]. URL: https://arxiv.org/abs/2312.11994 (cit. on p. 17).

[41] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. *FLAME: Free-form Language-based Motion Synthesis & Editing*. 2023. arXiv: 2209.00349 [cs.CV]. URL: https://arxiv.org/abs/2209.00349 (cit. on p. 18).

[42] Haowen Sun, Ruikun Zheng, Haibin Huang, Chongyang Ma, Hui Huang, and Ruizhen Hu. «LGTM: Local-to-Global Text-Driven Human Motion Diffusion Model». In: *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24*. SIGGRAPH '24. ACM, July 2024, pp. 1–9. DOI: 10.1145/3641519.3657422. URL: http://dx.doi.org/10.1145/3641519.3657422 (cit. on p. 18).

[43] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. «MoFusion: A Framework for Denoising-Diffusion-based Motion Synthesis». In: *Computer Vision and Pattern Recognition (CVPR)*. 2023 (cit. on p. 18).

[44] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. «PhysDiff: Physics-Guided Human Motion Diffusion Model». In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023 (cit. on pp. 18, 24).

[45] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. *T2M-GPT: Generating Human Motion from Textual Descriptions*. 2023. arXiv: 2301.06052 [cs.CV]. URL: https://arxiv.org/abs/2301.06052 (cit. on pp. 18, 24, 45).

[46] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. *MoMask: Generative Masked Modeling of 3D Human Motions*. 2023. arXiv: 2312.00063 [cs.CV]. URL: https://arxiv.org/abs/2312.00063 (cit. on pp. 19, 24).

[47] Wenfeng Song, Xinyu Zhang, Shuai Li, Yang Gao, Aimin Hao, Xia Hau, Chenglizhao Chen, Ning Li, and Hong Qin. «HOIAnimator: Generating Text-Prompt Human-Object Animations Using Novel Perceptive Diffusion Models». In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2024, pp. 811–820. DOI: 10.1109/CVPR52733.2024.00083 (cit. on p. 20).

[48] Xiaogang Peng, Yiming Xie, Zizhao Wu, Varun Jampani, Deqing Sun, and Huaizu Jiang. «HOI-Diff: Text-Driven Synthesis of 3D Human-Object Interactions using Diffusion Models». In: *arXiv preprint arXiv:2312.06553* (2023) (cit. on p. 20).

[49] Zhenzhi Wang, Jingbo Wang, Yixuan Li, Dahua Lin, and Bo Dai. *InterControl: Zero-shot Human Interaction Generation by Controlling Every Joint*. 2024. arXiv: 2311.15864 [cs.CV]. URL: https://arxiv.org/abs/2311.15864 (cit. on p. 20).

[50] Tenglong Ao, Zeyi Zhang, and Libin Liu. *GestureDiffuCLIP: Gesture Diffusion Model with CLIP Latents*. 2023. arXiv: 2303.14613 [cs.CV]. URL: https://arxiv.org/abs/2303.14613 (cit. on p. 20).

[51] Uttaran Bhattacharya, Elizabeth Childs, Nicholas Rewkowski, and Dinesh Manocha. «Speech2AffectiveGestures: Synthesizing Co-Speech Gestures with Generative Adversarial Affective Expression Learning». In: *Proceedings of the 29th ACM International Conference on Multimedia*. MM '21. New York, NY, USA: Association for Computing Machinery, 2021 (cit. on p. 20).

[52] Kiran Chhatre, Radek Dan??ek, Nikos Athanasiou, Giorgio Becherini, Christopher Peters, Michael J. Black, and Timo Bolkart. «Emotional Speech-driven 3D Body Animation via Disentangled Latent Diffusion». In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024, pp. 1942–1953 (cit. on p. 20).

[53] S. Nyatsanga, T. Kucherenko, C. Ahuja, G. E. Henter, and M. Neff. «A Comprehensive Review of Data-Driven Co-Speech Gesture Generation». In: *Computer Graphics Forum* 42.2 (May 2023), pp. 569–596. ISSN: 1467-8659. DOI: 10.1111/cgf.14776. URL: http://dx.doi.org/10.1111/cgf.14776 (cit. on p. 20).

[54] Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. «Speech gesture generation from the trimodal context of text, audio, and speaker identity». In: *ACM Transactions on Graphics* 39.6 (Nov. 2020), pp. 1–16. ISSN: 1557-7368. DOI: 10.1145/3414685.3417838. URL: http://dx.doi.org/10.1145/3414685.3417838 (cit. on p. 20).

[55] Lei Zhong, Yiming Xie, Varun Jampani, Deqing Sun, and Huaizu Jiang. *SMooDi: Stylized Motion Diffusion Model*. 2024. arXiv: 2407.12783 [cs.CV]. URL: https://arxiv.org/abs/2407.12783 (cit. on pp. 21, 24).

[56] Ian Mason, Sebastian Starke, and Taku Komura. «Real-Time Style Modelling of Human Locomotion via Feature-Wise Transformations and Local Motion Phases». In: *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 5.1 (May 2022). DOI: 10.1145/3522618 (cit. on p. 21).

[57] Paul Ekman and Wallace V Friesen. «Facial action coding system». In: *Environmental Psychology & Nonverbal Behavior* (1978) (cit. on pp. 23, 36, 41, 42).

[58] Unity Technologies. *Muse Animate*. Accessed: 2025-02-06. 2024. URL: https://docs.unity3d.com/Packages/com.unity.muse.animate@1.2/manual/index.html (cit. on p. 24).

[59] Sihao Wang, Xianmei Wang, Zhiliang Wang, and Ruoxiu Xiao. «Emotion Recognition Based on Static Human Posture Features». In: *6th International Technical Conference on Advances in Computing, Control and Industrial Engineering (CCIE 2021)*. Ed. by Yuriy S. Shmaliy and Abdelhalim Abdelnaby Zekry. Singapore: Springer Nature Singapore, 2022, pp. 529–539. ISBN: 978-981-19-3927-3 (cit. on p. 25).

[60] Catherine L. Reed, Eric J. Moody, Kathryn Mgrublian, Sarah Assaad, Alexis Schey, and Daniel N. McIntosh. «Body Matters in Emotion: Restricted Body Movement and Posture Affect Expression and Recognition of Status-Related Emotions». In: *Frontiers in Psychology* 11 (2020). ISSN: 1664-1078. DOI: 10.3389/fpsyg.2020.01961. URL: https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2020.01961 (cit. on p. 25).

[61] Abhishesh Homagain and Kaylena A. Ehgoetz Martens. «Emotional states affect steady state walking performance». In: *PLOS ONE* 18.9 (Sept. 2023), pp. 1–15. DOI: 10.1371/journal.pone.0284308. URL: https://doi.org/10.1371/journal.pone.0284308 (cit. on p. 25).

[62] Shihao Xu, Jing Fang, Xiping Hu, Edith Ngai, Wei Wang, Yi Guo, and Victor C. M. Leung. *Emotion Recognition From Gait Analyses: Current Research and Future Directions*. 2022. arXiv: 2003.11461 [cs.HC]. URL: https://arxiv.org/abs/2003.11461 (cit. on p. 25).

[63] Natalie Depraz. «Chapter 2. Shock, twofold dynamics, cascade: Three signatures of surprise. The micro-time of the surprised body». In: *Surprise at the Intersection of Phenomenology and Linguistics*. Ed. by Natalie Depraz and Agnès Celle. John Benjamins Publishing Company, 2019, pp. 23–42. ISBN: 9789027262424. DOI: doi:10.1075/ceb.11.02dep. URL: https://doi.org/10.1075/ceb.11.02dep (cit. on p. 26).

[64] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. «SMPL: A Skinned Multi-Person Linear Model». In: *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34.6 (Oct. 2015), 248:1–248:16 (cit. on p. 29).

[65] Ellen Blythe, Lúcia Garrido, and Matthew R. Longo. «Emotion is perceived accurately from isolated body parts, especially hands». In: *Cognition* 230 (2023), p. 105260. ISSN: 0010-0277. DOI: https://doi.org/10.1016/j.cognition.2022.105260. URL: https://www.sciencedirect.com/science/article/pii/S0010027722002487 (cit. on pp. 40–43, 45, 46).

[66] Laura Martinez, Virginia B Falvello, Hillel Aviezer, and Alexander Todorov. «Contributions of facial expressions and body language to the rapid perception of dynamic emotions». In: *Cognition and Emotion* 30.5 (2016), pp. 939–952 (cit. on pp. 40–42, 46).

[67] Aline Normoyle, Fannie Liu, Mubbasir Kapadia, Norman I Badler, and Sophie Jörg. «The effect of posture and dynamics on the perception of emotion». In: *Proceedings of the ACM symposium on applied perception*. 2013, pp. 91–98 (cit. on pp. 40–43, 45).

[68] Anthony P Atkinson, Winand H Dittrich, Andrew J Gemmell, and Andrew W Young. «Emotion Perception from Dynamic and Static Body Expressions in Point-Light and Full-Light Displays». In: *Perception* 33.6 (2004). PMID: 15330366, pp. 717–746. DOI: 10.1068/p5096. eprint: https://doi.org/10.1068/p5096. URL: https://doi.org/10.1068/p5096 (cit. on pp. 40–42, 45, 46).

[69] Magzhan Mukanova, Nicoletta Adamo, Christos Mousas, Minsoo Choi, Klay Hauser, Richard Mayer, and Fangzheng Zhao. «Animated Pedagogical Agents Performing Affective Gestures Extracted from the GEMEP Dataset: Can People Recognize Their Emotions?» In: *ArtsIT, Interactivity and Game Creation*. Ed. by Anthony L. Brooks. Cham: Springer Nature Switzerland, 2024, pp. 271–280. ISBN: 978-3-031-55312-7 (cit. on p. 48).

[70] Klara Brandstätter, Ben J. Congdon, and Anthony Steed. «Do you read me? (E)motion Legibility of Virtual Reality Character Representations». In: *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 2024, pp. 299–308. DOI: 10.1109/ISMAR62088.2024.00044 (cit. on p. 48).

[71] Jonas Oppenlaender, Rhema Linder, and Johanna Silvennoinen. «Prompting AI art: An investigation into the creative skill of prompt engineering». In: *International journal of human–computer interaction* (2024), pp. 1–23 (cit. on p. 50).