

ReNeLiB: Real-time Neural Listening Behavior Generation for Socially Interactive Agents

Daksitha Withanage Don
University of Augsburg,
German Research Center for Artificial
Intelligence
Augsburg, Germany,
daksitha.withanage.don@uni-a.de

Philipp Müller
German Research Center for Artificial
Intelligence
Saarbruecken, Germany
philipp.mueller@dfki.de

Fabrizio Nunnari
German Research Center for Artificial
Intelligence
Saarbruecken, Germany
fabrizio.nunnari@dfki.de

Elisabeth André
University of Augsburg
Augsburg, Germany
andre@informatik.uni-augsburg.de

Patrick Gebhard
German Research Center for Artificial
Intelligence
Saarbruecken, Germany
patrick.gebhard@dfki.de

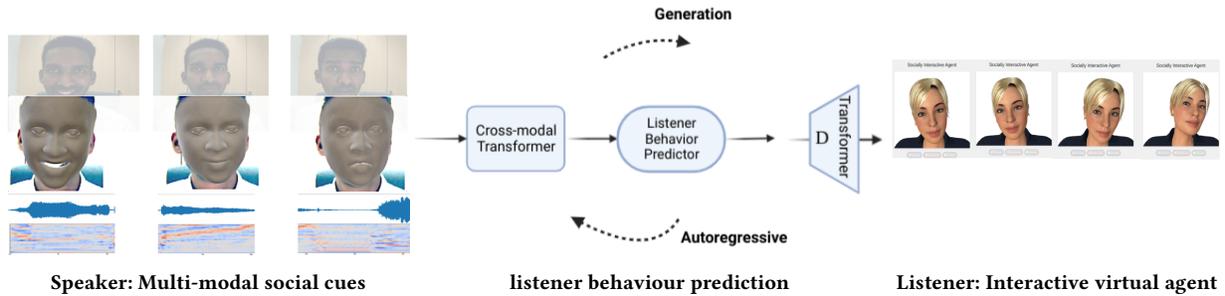


Figure 1: Experience ReNeLiB: A framework transforming human-webcam interactions through 3D motion and Mel frequency analysis, enabling interactive virtual agents to adapt to user behavior based on multimodal social cues.

ABSTRACT

Flexible and natural nonverbal reactions to human behavior remain a challenge for socially interactive agents (SIAs) that are predominantly animated using hand-crafted rules. While recently proposed machine learning based approaches to conversational behavior generation are a promising way to address this challenge, they have not yet been employed in SIAs. The primary reason for this is the lack of a software toolkit integrating such approaches with SIA frameworks that conforms to the challenging real-time requirements of human-agent interaction scenarios. In our work, we for the first time present such a toolkit consisting of three main components: (1) real-time feature extraction capturing multi-modal social cues from the user; (2) behavior generation based on a recent state-of-the-art neural network approach; (3) visualization of the generated behavior supporting both FLAME-based and Apple ARKit-based interactive agents. We comprehensively evaluate the real-time performance of the whole framework and its components. In addition, we introduce pre-trained behavioral generation models derived from psychotherapy sessions for domain-specific listening behaviors. Our software toolkit, pivotal for deploying and assessing SIAs' listening behavior in real-time, is publicly available. Resources, including code, behavioural multi-modal features extracted from therapeutic interactions, are hosted at <https://daksitha.github.io/ReNeLiB>

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI)**.

1 INTRODUCTION

Socially-aware Interactive Agents (SIAs) are autonomous systems proficient in engaging in natural language dialogues and interacting with their environment [30]. The increasing importance of human-machine interaction in everyday life necessitates the development of SIAs that can actively listen and respond to users in a believable and context-dependent manner. The Media Equation theory posits that individuals treat computers, televisions, and new media similarly to real people and places [32], suggesting that SIAs displaying social behaviors, including active listening, can positively impact user experiences. It is essential to argue that SIAs should operate in a manner tailored to the context in which they are deployed. This emphasizes the need for SIAs to demonstrate context-sensitive, believable, and professional behavior. By doing so, these agents can create more immersive and natural interactions, ultimately leading to improved human-SIA communication and user satisfaction.

Facial expressions and head movements are essential components of human communication and social interaction, extending their importance to human-SIA interactions. Bickmore and Cassell [3] evaluated an embodied conversational agent (ECA) that

used nonverbal cues, such as nodding and eyebrow movements, to express active listening. Their findings revealed that users perceived the ECA as more attentive and engaging compared to a version without these social cues. Context comprehension and accounting for the multimodal nature of interactions are crucial, as facial gestures and spoken utterances are intrinsically interconnected. For instance, in psychotherapy, nonverbal synchrony has proven vital, with head synchrony positively correlating to therapy success [31]. Consequently, incorporating context-dependent facial gestures in active communicative listening behavior is imperative for fostering natural human-SIA interactions.

Traditional authoring and assistive frameworks for SIAs have often relied on predefined scripts to generate non-verbal behavior, playing a crucial role in the development of conversational agents and virtual characters. For example, the BEAT toolkit enables the automatic generation of gestures and facial expressions based on input text [6]. Moreover, the Behavior Markup Language (BML) provides a unified framework for generating and controlling multimodal behavior in virtual agents, including facial expressions, gestures, and gaze direction [20]. Additionally, Pelachaud’s work on modeling multimodal emotional expression in virtual agents has contributed to the understanding of how predefined scripts can be used for generating non-verbal behaviors [29]. However, these methods may be limited in capturing the natural complexity and dynamic nature of human-human non-verbal communicative behavior, necessitating further research on alternative techniques that can address these limitations.

The advent of deep learning techniques has facilitated data-driven generative approaches for creating locomotion, dancing, and facial gestures in avatars [18, 26]. Despite their potential, these techniques exhibit limited real-time dyadic capabilities, rendering them unsuitable for direct implementation in SIAs. The importance of real-time capabilities in practical applications is critical for fostering immersive and engaging human-SIA interactions [37]. As such, there is a pressing need to address these limitations and develop a framework using data-driven generative methods that can meet the real-time dyadic requirements of SIAs, enhancing the naturalness and efficacy of human-SIA communication across various contexts.

In this paper, we present a novel open-source modular software toolkit designed to overcome these limitations by facilitating the integration of data-driven behavior generation with multiple facial parametric representations. Our approach employs the FLAME (Faces Learned with an Articulated Model and Expressions) [22], a highly expressive 3D face parametric representation. To accommodate other facial parametric representations, we devise a mapping function for Apple Inc.’s ARKit ARFaceAnchor [16]. Our toolkit incorporates state-of-the-art generative models for real-time non-verbal active listening behaviors in dyadic SIA interactions, dynamically adjusting listener behavior according to conversational context. This framework bridges the gap in existing approaches by enabling real-time, multimodal feature representation and seamless integration of data-driven models. Furthermore, we present a method to enhance the expressiveness of industrial-standard SIAs using the commercial VuppetMaster platform, developed by Charamel [11]. This is achieved by establishing a coherent transformation between FLAME expression coefficients and Apple ARKit expressions, enabling seamless integration of generated behavioral

expressions within the VuppetMaster platform. By offering an open-source framework for data-driven listener behavior generation, our work paves the way for the development of increasingly sophisticated SIAs and their applications in a wide range of contexts.

2 RELATED WORK

Our work is related to interactive virtual agents, recent approaches to data-driven behavior generation, and representation systems for human visual social behavior.

2.1 Interactive Virtual Agents

Interactive Virtual Agents (IVAs) aim to generate dynamic social behaviors and maintain user engagement in real-time, fully dyadic conversations [10, 24, 25]. The significance of interactional motion within conversational agents has been increasingly recognized, as it enhances user engagement and fosters more natural communication experiences. Studies have investigated rapport building in virtual agents [12, 14], the impact of animated conversations on user experience [5, 25], and the importance of situated interaction in IVAs [4]. Additionally, Gebhard and colleagues [10] explore the role of gestures and body language in improving communication, proposing a conversational flow for real-time, fully dyadic interactions.

Nonetheless, the majority of prior research has focused on rule-based methods, employing motion capture sequences or hand-crafted animations for interactional motion in facial gestures and speech [4, 10, 25]. These rules encompass gaze behavior [17], turn-taking management [14], facial expressions [27], gestures and body language [6], and backchanneling [39]. These approaches exhibit limitations regarding the range of captured gestures and the simplifying assumptions made for motion generation, rendering them less suitable for context-dependent dynamic interactions.

2.2 Data-driven Approaches for Behavioral Animation Synthesis

In response to the limitations of rule-based methods, recent studies have explored data-driven approaches for generating conversational behavior in IVAs while leveraging large datasets and advanced modeling techniques to capture the subtleties of human behavior. For example, [21] investigated machine learning techniques and deep learning models to create more contextually relevant and natural speaker behaviors. Extending the automatic locomotion synthesis framework MoGlow [13], the "Let’s face it" [18] study devised a probabilistic approach for synthesizing facial gestures that account for interlocutor awareness in dyadic conversations. However, this method’s efficacy was constrained by not differentiating between speech-related and non-speech-related facial gestures during feature extraction. In light of the interdependence between speech and gesture perception, the "Learn2Listen" method employed transformer-based VQ-VAE and multimodal fusion techniques [26] to predict non-verbal facial behavior, yielding promising results in capturing facial gesture nuances and efficient behavior generation. Despite these advancements, the integration of data-driven approaches into real-time, fully dyadic conversational flows remains a challenge due to the lack of an appropriate framework

capable of handling the computational requirements, synchronizing multi-modal inputs, and facilitating seamless integration with existing IVA systems [10, 24].

2.3 Behaviour Representation

Advancements in photorealistic virtual humans have led to more believable and engaging representations, significantly contributing to the development of intelligent virtual agents [33]. A primary challenge in this domain, however, is the lack of standardization in agent animation, which impedes progress in automatically generating realistic agent behavioral animation using data-driven approaches. Different systems utilize various techniques, such as blendshape and bone animation, to define their animation controllers [20]. Additionally, creating photorealistic virtual humans for real-time IVAs demands considerable expertise and resources due to the process’s inherent complexity [23].

Van der Struijk and colleagues [38] employed the 3D human model from the open-source FACSHuman¹ software add-on to drive facial motors in real-time using the Facial Action Coding System (FACS) detected with OpenFace [2]. However, their approach focused on mimicking facial behavior rather than generating social behavior based on the interactive context in Human-Computer Interaction setups using IVAs. A limitation of using FACS for facial action unit representation is its inability to effectively capture subtle expressions and head rotations [18]. Furthermore, [38] highlighted the limitations of OpenFace [2] in detecting FACS Action Units (AUs) and intensity values, as its AU and intensity value predictors are not synchronously trained, leading to inaccuracies. As a result, manual post-processing was necessary in [38] to fine-tune the intensity value for activating facial animation.

ARFaceAnchor² has been developed to enable real-time face tracking systems on native devices. Projects such as [9] have utilized ARKit to animate socially interactive agents in real-time. However, using ARKit introduces significant drawbacks due to its device dependencies, as the facial 3D mesh cannot be employed outside the native platform. This limitation hinders the extraction of facial expressions and head movements from large video datasets recorded with monocular cameras, which are essential for training generative machine learning models like [18, 26].

Researchers have explored open-source alternatives such as RingNet [34], which learns to regress 3D face shape and expression from an image without 3D supervision, offering functionality comparable to ARFaceAnchor yet with different animation controllers. However, RingNet’s complex neural network architecture results in substantial computational expense [26]. DECA [8], an improved version of RingNet, leverages the FLAME model [22] and a convolutional neural network for efficiently capturing and animating 3D facial expressions from single 2D images. Despite its ability to generate more realistic facial reconstruction, real-time processing remains challenging due to DECA’s face-alignment module, which causes a computational bottleneck. EMOCA [7] extends DECA’s implementation to improve 3D facial reconstruction with higher

emotional fidelity, employing a deep perceptual emotion consistency loss during training. This novel approach outperforms existing methods in expression quality and perceived emotional content, demonstrating the potential of 3D geometry, yet it has not been extended in reconstructing 3D facial representations for real-time face tracking and 3D morphing. We extended this method to work in real-time 3D morphing and develop a novel facial expression transformation, that aims to bridge the gap between data-driven techniques and commercially available IVAs, fostering seamless integration and improved human-computer interaction experiences.

3 FRAMEWORK

Human-to-human communication involves a complex interplay of verbal and nonverbal cues. Furthermore, the behavior of a listener during a conversation can depend on the context of the conversation and the social setting, as evidenced by the nuances observed in human-to-human communication and real-life therapeutic interactions, as shown in Fig. 1.

Our main objective is to develop a framework that predicts a socially interactive agent’s listening facial behavior in real-time based on the user’s multimodal social cues. Specifically, we aim to predict the interactive agent’s facial expressions, \hat{F}_t^{sia} , at each time-step t , given the user’s contextual speaking information encapsulated in audio features $A_{1:t}^{user}$ and facial features $F_{1:t}^{user}$, and any past predicted facial behavior for the agent to process them in an autoregressive manner, predict w listener behavioral sequence $F_{t:t+w}^{sia}$. Therefore, we model the distribution P of the agent’s predicted facial behavior, learned from the therapist’s listening behavior \hat{F}_t^{thera} conditioned on audio features from the patient $A_{1:t}^{pat}$ and facial features $F_{1:t}^{pat}$ taking into account the patient’s multimodal contextual information. Therefore, we model the distribution P of the interactive agent’s predicted listening behavior, learned \mathcal{L} from the therapist’s listening behavior, as:

$$P(\hat{F}_t^{sia} | A_{1:t}^{user}, F_{1:t}^{user}) = \mathcal{L}(P(\hat{F}_t^{thera} | A_{1:t}^{pat}, F_{1:t}^{pat})) \quad (1)$$

Our framework then utilizes P distribution for predicting a socially interactive agent’s (SIA) listening facial behavior with the user’s multimodal social cues processed in real-time. The framework employs deep learning techniques to model the relationship between user’s audio and facial features and the agent’s facial behavior, using a dataset of real-life therapeutic video data recordings for training. Comprising several interconnected modules, such as the User with Voice Activity Detector, Real-time Feature Extraction, Behavior Generator, FASTApi Server with Local-to-Global Transformation, and LiveFLAME Blender Add-on, the framework is designed for high throughput and real-time processing supporting both FLAME-based and ARKit-based interactive agents. By leveraging the ZeroMQ³ distributed messaging library and web sockets for communication, the framework aims to enhance the expressiveness and responsiveness of listening behavior for the SIA in various applications, such as telemedicine, mental health counseling, and customer support.

¹<https://www.michaelgilbert.fr/facshuman/>

²<https://developer.apple.com/documentation/arkit/ifaceanchor/blendshapelocation>

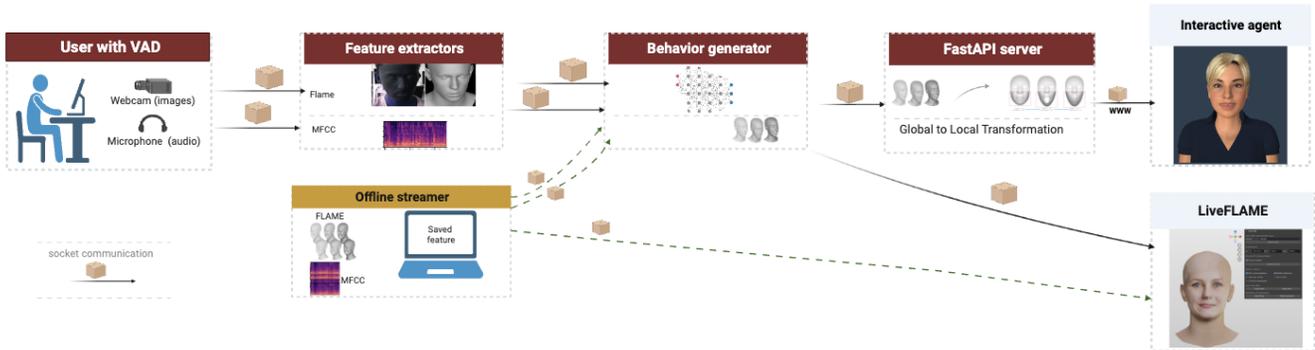


Figure 2: Overview of the real-time interactive framework modules, utilizing a publisher-subscriber pattern to enable real-time listener behavior. The framework supports both online and offline modes, allowing for feature extraction from sensors or streaming from locally saved files. The solid arrows represent the modules used in our evaluation.

3.1 Framework Modules

The proposed framework, depicted in Fig. 2, comprises several interconnected modules that collaborate to generate the listening facial behavior of a SIA in real-time. The framework has been designed to support both online processing, using sensor data, and offline processing, which allows from prior extracted feature data. In this section, we present a comprehensive description of each module. The modules work together to enable the SIA to exhibit realistic facial behavior.

- (1) **User with Voice Activity Detector** is responsible for detecting when the user is speaking. Recognizing voice activity activates the listener behavior for the SIA, enabling more natural and responsive interactions between the user and the agent.
- (2) **Real-time Feature Extraction** captures and processes audio and facial features, specifically FLAME and MFCC features, synchronously. Proper alignment between the extracted features is ensured using timestamps, which helps maintain accurate temporal information for the input data.
- (3) **Behavior Generator** employs a producer-consumer pattern using fixed-length double-ended queues for efficient data handling in a multi-threaded environment. This design allows multimodal producers to write data to the queues, while a consumer thread processes the data within a sliding window, predicting facial behaviors with an adjustable processing rate and publishing the data to the subsequent module in the pipeline.
- (4) **FASTapi Server with Local-to-Global Transformation** serves as the back-end for the web-based IVA front-end. It streams data to a IVA such as VuppetMaster⁴ or MetaHuman [9], processing rotational and facial expression transformations such as FLAME to ARKit.
- (5) **Interactive Virtual Agent** is a web-based plugin powered by VuppetMaster, designed for the integration into HTML web browsers. This could be adjustable to work with another character animator like MetaHuman [9].

- (6) **LiveFLAME Software Add-on** is a real-time visualizer of FLAME parameters within the Blender⁵ software. Implemented as a Blender add-on, it enables users to monitor and evaluate the generated facial behavior by directly mapping the predicted FLAME parameters to a FLAME-based facial model (e.g., female or male). This visualization tool provides valuable insights during the development and testing of the framework.

4 IMPLEMENTATION

In this section, we discuss the implementation of the proposed system, also including an evaluation of the trained behavior generative models and the dataset utilized.

4.1 Facial and Audio Representations

Our framework employs the FLAME statistical 3D head model [22] and the EMOCA face reconstruction framework [7] to represent facial expressions and head movements. The FLAME model consists of three critical components: expression, pose, and shape vectors. The expression vector $\psi_e \in \mathbb{R}^{1 \times 100}$ captures variations in facial expressions, encoding facial muscle movements within a reduced-dimensional space derived from a 3D scanned dataset using Principal Component Analysis (PCA). The pose vector $\theta \in \mathbb{R}^{1 \times 15}$ represents head orientation and rotations of specific joints (e.g., neck, jaw, eyeballs), describing the overall position and orientation of the 3D head model in 3D space. The shape vector $\beta \in \mathbb{R}^{1 \times 300}$ encodes individual facial identity, encompassing the unique structure and geometry of the face. This vector defines the base 3D head model, which is subsequently modified by the expression and pose vectors to create the final 3D head representation.

For prosodic behavior modeling, we extracted Mel Frequency Cepstral Coefficients vector $MFCC = c_1, c_2, \dots, c_l$ from the audio as an audio representation, where l represents the number of coefficients. MFCCs capture phonetic information and provide a compact representation of the audio signal, making them suitable for real-time applications with compact data footprints for social cues via audio. Moreover, MFCCs have been extensively employed

³<https://zeromq.org/>

⁴<https://www.charamel.com/en/software/vuppetmaster>

⁵www.blender.org

in speech and emotion recognition tasks [36], demonstrating their efficacy in capturing relevant information from audio signals.

4.2 Real-life Therapy Interaction Dataset

In recent years, the development of virtual therapists has garnered significant interest, with the aim of providing mental health support through digital platforms. However, obtaining access to real-life therapeutic video data, to train data-driven generative models, remains a challenge due to the sensitive nature of such interactions. To address this issue, we collaborated with [28, 35] to acquire real-life therapeutic video data recordings. Further details about the data are listed in Appendix A.

Given the sensitive nature of patient-therapy interactions, it is crucial to ensure the confidentiality of the data. As such, we employed a Secure Machine Learning Architecture (SEMLA) [1] for data processing and machine learning model training. Our primary objective was to process separate video streams and audio channels for both patients and therapists, in order to create a feature dataset for model training.

4.3 Conditional Motion Synthesis of Conversational Dynamics

This section outlines the training and evaluation processes of unsupervised machine learning models designed for conditional motion synthesis between a speaker and a listener. The approach by Ng et al. [26] serves as our foundation. We extend the original method in two significant ways. Firstly, we replace the DECA 3D Morphable Face Model [8] with EMOCA [7] to estimate 3D facial expressions. Secondly, we used the pyanote speaker-diarization method⁶ for identifying and separating speakers during interactions. Following the methodology in [26], our learning task is represented in Eq. 1. We then proceed with model training.

4.3.1 Model Training. Data were derived from sessions conducted by "TherapistA". This segmentation resulted in intervals: $S_{\text{backchanneling}}$, $S_{\text{short-speech}}$, and $S_{\text{long-speech}}$. Additional details regarding speech activity segmentation are presented in Appendix B. The TherapistA dataset spans 12 hours, recorded at 25 fps, presenting split video recordings with the therapist positioned to the right and the patient to the left. The model training process encompassed an initial phase of VQ-VAE pre-training and the Predictor module is trained separately. The Predictor model is excluded from back-propagation during E, Z, and D training. The notation employed in this section is consistent with that found in [26].

Pre-training the VQ-VAE: Our approach employs a transformer-based encoder (E) and decoder (D) architecture to efficiently capture the therapist’s facial motion and expression from video data, as illustrated in Fig. 3. The training process involves the encoder, decoder, and codebook (Z) components, utilizing the loss function described in [26]. The dataset is partitioned into 70% training, 20% validation, and 10% testing portions. The Adam optimization algorithm is used to fine-tune the model’s parameters with the goal of minimizing the loss function. After approximately 5,000 training steps, the best-performing model on the validation set is preserved for the next phase.

⁶<https://github.com/pyanote/pyanote-audio>

Training the Predictor Module: Upon pre-training the VQ-VAE, the encoder and codebook components are kept fixed, and the focus shifts to training the transformer-based predictor module, as depicted in Fig. 4. The predictor is designed to learn temporally long-range patterns in the input sequence by employing cross-modal attention to fuse audio features and facial motion features as the conditioning vector. This process is coupled with the discretized past listener motion sequence encoding provided by the pre-trained encoder. The autoregressive predictor outputs a distribution over the $K = 200$ discrete codebook indices, from which a code for the subsequent timestep is sampled and then passed to the trained decoder. We selected values $T = 64$, $t = 32$, and $w = 8$, along with a Mel frequency length of $l = 128$, similar to [26], to evaluate our model with their pre-trained model.

We applied the model trained on TV interviewer Conan, as established by [26], which encompassed a range of participants in interviews. To evaluate the TherapistA model, a comparative matrix was devised, incorporating both its ground truth data and the ground truth from interviewees interacting with Conan. The same assessment was performed for the Conan model using the ground truth data of patients and interviewees. We calculated L2 loss to ensure compatibility with the reference study [26] while their baseline model achieved an L2 loss of 52.68.

Table 1: L2 Loss values for machine learning models tested on different datasets

Dataset	Therapist	Interviewer
	L2 Loss	L2 Loss
Patients	41.06	89.49
Interviewee	78.85	59.72

4.3.2 Discussion. It is important to recognize that the L2 loss accentuates disparities between predicted and ground truth temporal FLAME vector sequences, leading to larger error values for substantial deviations. The results, presented in Table 1, should be interpreted with the understanding that there are language and interaction context differences between the two datasets. This highlights the significance of behavior modeling within the interaction context.

4.4 Real-time Framework Application

The models that we developed predict behavior in the FLAME head-pose and expression format, necessitating a FLAME-compatible interactive agent for visualization. We utilize the LiveFLAME software add-on as described in Section 3.1. However, given the rapid advancements in photorealistic socially interactive agents [9, 11, 33], each with their unique facial animation systems, our objective is to enhance the compatibility of our synthesized behaviors with industry-standard socially interactive agents, such as VuppetMaster [11]. Notably, Charamel [11] supports the research community by providing IVAs for collaborative projects. VuppetMaster’s visually appealing full-body agents (see the agent in the upper right of Figure 3) are animated using VuppetMaster’s animation engine and deployed as a web-based solution. This engine is based on a

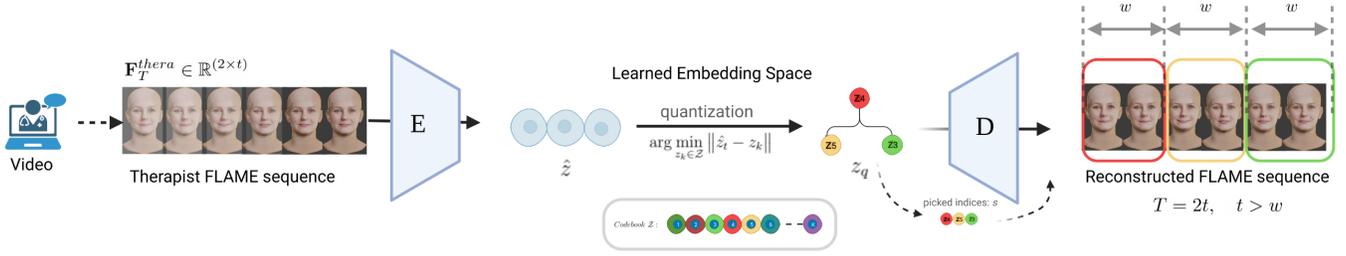


Figure 3: VQ-VAE training process

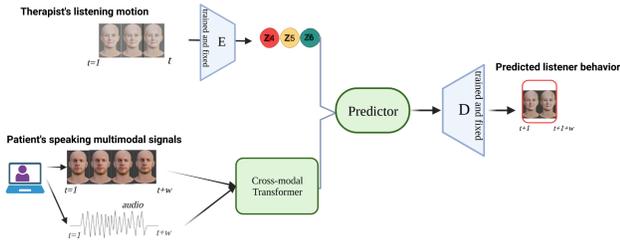


Figure 4: Predictor training process

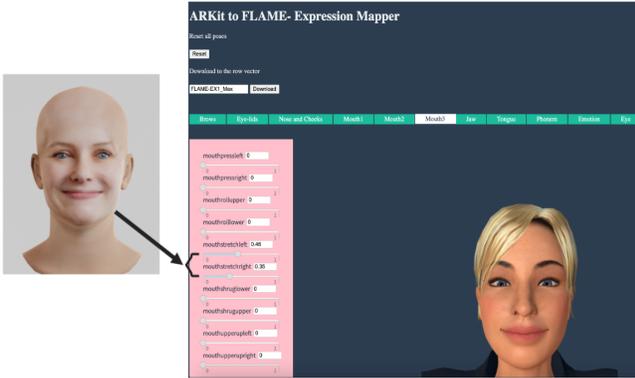


Figure 5: FLAME to ARKit expression mapper

humanoid facial skeletal structure incorporating facial muscle controllers, which enable a comprehensive range of facial expressions and movements. The controllers' naming scheme and shape manipulation capabilities are designed to be compatible with ARKit facial muscle controllers [16]. However, it should be noted that the facial muscle controllers in the FLAME mesh follow a distinct representation, necessitating a suitable conversion process for seamless integration.

4.4.1 Constructing the Global-to-Local Transformation Matrix. In this work, we introduce a novel real-time linear transformation designed for efficient execution in real-time applications. While the FLAME model employs global expressions, simultaneously controlling multiple facial muscles, ARKit utilizes local expressions that focus on individual muscle control, enabling more localized

facial movements. We aim to develop a computationally efficient transformation matrix for real-time scenarios. To facilitate the transformation between FLAME expression coefficients and Apple ARKit facial expressions, a global-to-local transformation matrix \mathcal{GL} is constructed. Furthermore, the jaw and head poses are independently converted from axis angle representation to Euler angles.

The row vectors of \mathcal{GL} are derived by mapping extreme FLAME expressions using multiple corresponding ARKit expressions using an expression mapper, as shown in Fig. 5. This process is iterated for selected $\pm F_e \in \mathbb{R}^{\|\psi_e\| \times 2}$ FLAME expressions, where the "+-" in the equation indicates the extremes (-3 and 3) of each FLAME expression.

$$\mathcal{GL} = \begin{bmatrix} \vdots & \vdots & \vdots & \ddots & \vdots \\ \pi_e(i, 1) & \pi_e(i, 2) & \pi_e(i, 3) & \cdots & \pi_e(i, 52) \end{bmatrix} \quad (2)$$

4.4.2 Real-Time Transformation of FLAME Vectors to ARKit Vectors. Given a batch size n of FLAME vectors \mathcal{F} , the transformed to $n \times 52$ ARKit expression matrix \mathcal{A} computed using the transformation matrix:

$$\mathcal{A}(n \times 52) = \text{normalise}(\mathcal{F}(n \times F_e) \times \mathcal{GL}(F_e \times 52), 0, 1) \quad (3)$$

For the conversion of rotation angles, the following operations are performed:

$$\begin{aligned} \mathbf{q}_{\text{jaw}} &= \text{axisAngleToQuaternion}(\boldsymbol{\alpha}_{\text{jaw}}), \\ \mathbf{q}_{\text{head}} &= \text{axisAngleToQuaternion}(\boldsymbol{\alpha}_{\text{head}}), \\ \mathbf{e}_{\text{jaw}} &= \text{quaternionToxyzEuler}(\mathbf{q}_{\text{jaw}}), \\ \mathbf{e}_{\text{head}} &= \text{quaternionToxyzEuler}(\mathbf{q}_{\text{head}}), \end{aligned} \quad (4)$$

5 EVALUATION

In our system, real-time operation is achieved through parallel processing and a modular architectural approach. We adopt the runtime evaluation metrics proposed by "Facsvatar" [38]. Video features, represented by FLAME, and audio features, denoted by MFCCs, are extracted concurrently. This simultaneous extraction ensures seamless streaming of features using ZeroMQ, with timestamps to guarantee synchronization. We conducted evaluations of our system on both high-end (13th generation Intel Core i9 and GeForce RTX 4090) and mid-range (Intel Core i7 and RTX 1080) configurations. Notably, while individual modules, such as Mel and

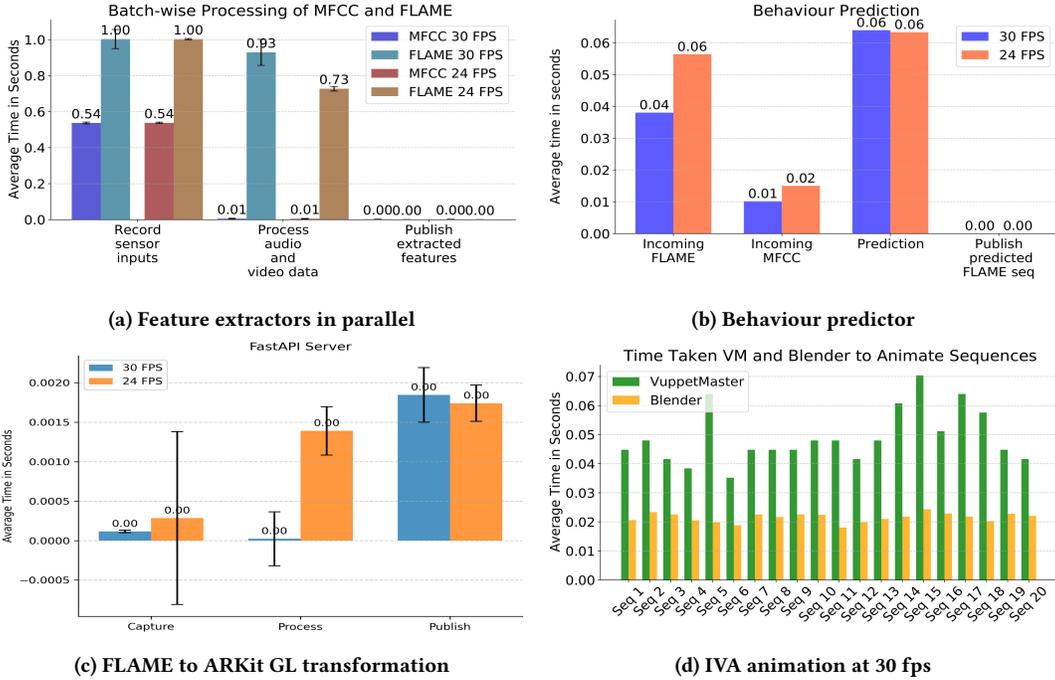


Figure 6: Network latency plot for each framework module for different frame rates

FLAME for behavior generation, demand processing time, the system’s intrinsic parallelism significantly mitigates the latency from a user’s input to the agent’s response. In this study, we present results from a test scenario conducted on an i9 platform running Ubuntu 22.04. We thoroughly assessed the performance of the ReNeLiB framework across various components.

In our design, each module employs a multi-threaded custom producer-consumer network architecture, using either ZeroMQ or WebSocket, to ensure independent data processing. This architecture utilizes a publisher-subscriber communication pattern, optimizing module-specific concerns and enabling flexible data processing speeds. The MFCC extractor adjusts to audio recording duration T_{audio} and sampling rate R , and publish the data to the behaviour predictor at M_{fps} . Concurrently, the FLAME extractor record video from a webcam for a duration T_{video} , batch-processes, and then publish at F_{fps} .

The behavior generation module receives these multimodal data streams parallelly to predict the next sequence of behavioral animation. LiveFLAME visualizer subscriber to this behavior predictor module directly as it can visualise predicted flame sequences. FastAPI backend handling facial motion transformation and publish to the VuppetMaster receives data from the backend to animate the ARKit-based VIA. Throughout this system, each module maintains low latency in its data receiving data, processing, and publishing data. Additionally, we quantified the time taken by the interactive agents to animate 32 behavioral sequences, as depicted in Fig. 6(d).

Performance Results: Using a camera operating at both 30 fps and 24 fps, coupled with a four-microphone array processing 16-bit audio at 16 kHz, our system was configured for optimal

performance. We set $T_{audio} = 0.5s$ for audio capture and subsequent Mel frequency processing. For video, $T_{video} = 1s$ was designated, allowing the FLAME extractor to batch-process and publish 30 or 24 images every second. The audio Mel frequencies were re-sampled at $4 \times F_{fps} \times T_{audio}$ fps, ensuring consistent feature processing. We have selected $F_{fps} = 30$ and $M_{fps} = 120$, regardless of capturing image rate. The results are illustrated in Fig. 6.

6 DISCUSSION

We present the first framework that allows for real-time interaction with virtual avatars driven by deep learning-based behavior generation. As such, we address a crucial need that is pointed out in recent publications on neural behavior generation, namely the lack of a possibility to evaluate, develop, and apply such architectures in interactive human-agent scenarios [18, 26].

6.1 On Performance

The results, as depicted in Fig. 6(a), elucidate the performance metrics of each module in our framework. The Behavior generator module showcased a swift processing time of 0.06 seconds, underscoring its real-time behavior generation efficiency. The Webcam FLAME extractor module registered batch-wise processing times of 0.79 and 0.83 seconds, contingent upon image quantity for 3D reconstruction, marking the highest computational latency. Still, this latency remains conducive for real-time applications as the input latency is higher than the output throughput. A noteworthy initial delay of 1s, highlighted in Fig.6 (a), arises from batch processing of audio-video data. This latency is primarily attributed to T_{video} , suggesting potential reductions by optimizing its value, though it’s

computationally demanding. While the FLAME processing could match camera frame rates, real-time performance was not feasible on an i7 machine. Consequently, we opted for batch-wise processing at 1s intervals. Despite an initial 1 s delay, our approach reliably delivers outputs at either 30 fps or 24 fps.

The Mel frequency extractor demonstrated a swift processing time of 0.01 seconds, with a capture window rate of 0.00054 per frame, underscoring its real-time audio data processing efficiency as depicted in Fig. 6(a). The FastAPI server with \mathcal{GL} transformation module clocked in at 0.0015 seconds for processing and 0.0019 seconds for publishing, highlighting its rapid data transformation and transmission capabilities. Fig. 6 showcases the animation speeds for virtual agents, with real-time performance ranging between 20 ms to 50 ms. The animation frame rate is modifiable in our framework to achieve realistic transformed facial motions.

Interpreting these results necessitates an understanding of system-specific processing variability. Our framework emphasizes real-time suitability in interactive settings by ensuring low latency and effective data processing. The adjustable delays (T_{video} and T_{audio}) derive from the method proposed in [26], which requires a 32 video frame sequence for output. Adapting to a 1s T_{video} for cameras at 30 or 24 fps, we’ve modified the overlapping stride of FLAME and MFCC features in the behavior predictor. When the camera fps deviates from the trained 32-frame video sequence and 4×32 for MFCC as per [26], we incorporate features from preceding batches for the L2L prediction, ensuring modality compatibility. This adaptability lets our system handle varying fps while delivering uninterrupted output.

6.2 Applications

The proposed real-time framework offers a versatile solution for applications demanding interactive and immersive experiences. Potential use cases include virtual assistants, teleconferencing, educational and training platforms, and healthcare settings such as telemedicine, and virtual consultations. The framework’s capacity to capture and represent users’ facial expressions and speech enable the development of engaging, realistic interactive agents, fostering enhanced user experiences and facilitating more effective human-computer interactions across various domains. While our framework is agnostic to the concrete application scenario, we provide pre-trained models that can be valuable to users. We include a model trained on psychotherapy interactions as these interactions are (1) rich in social cues and interpersonal synchronization, and (2) difficult to obtain by most researchers due to data privacy considerations. We opted to train a model for a single specific therapist to represent nuanced individual behavior instead of a model interpolating between different persons. This is in line with the Learning to Listen approach presented by Ng and colleagues [26], who trained individual models for TV presenters. ReNeLiB is designed for modularity, adaptability, and universality, catering to diverse virtual agents and platforms. While our behavior generation module utilizes the approach from [26], its modular design ensures compatibility with alternative behavioral prediction techniques, such as [18].

6.3 Limitations and Future Work

The current framework exhibits certain limitations, such as the VuppetMaster character animation, which is managed by iteratively setting expression key values and head rotation values through web-based JavaScript. For more fluid movements, it would be advantageous to transmit a sequence of animations directly to the VuppetMaster engine, necessitating collaboration with the developers of VuppetMaster. Moreover, the system generates 32 frames of animation sequences per second in an autoregressive manner; however, it does not interpolate between predicted animation sequences to achieve smoother behavior, presenting an opportunity for enhancement. In future work, the framework will undergo evaluation and user studies to assess the contextual appropriateness of generated behavior and to develop a standardized platform for evaluating listener behavior in interactive agents. These efforts will not only validate the framework’s effectiveness but also contribute to its enhancement, allowing it to better accommodate a diverse array of intelligent IVA.

ACKNOWLEDGMENTS

We would like to express our heartfelt gratitude to Prof. Dr. Cord Benecke and the diligent team at the University of Kassel, Germany, for their indispensable support. Their efforts, particularly in recording the therapy sessions, significantly helped our research. It is essential to note that the machine learning trained in our work do not retain any personal or health-related information. The anonymized data was utilized solely to derive facial behaviors, expressions, and raw audio features using MFCC. We strictly refrain from sharing any video or audio data with external entities and computation were conducted with Secure Machine Learning Architecture [1] specialised for sensitive data processing. This research has been generously supported by the German Federal Ministry for Education and Research (BMBF) as a segment of the UBIDENZ project, under grant number 13GW0568D. Further, P. Müller’s contributions were funded by the BMBF under grant number 01IS20075.

REFERENCES

- [1] Jan Alexandersson, Jochen Britz, Valentin Seimetz, and Daniel Tabellion. [n. d.]. White paper – SEMLA. <https://semlda.dfki.de/white-paper/>. (Accessed on 04/29/2023).
- [2] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. 2018. OpenFace 2.0: Facial Behavior Analysis Toolkit. In *2018 13th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2018)*. 59–66. <https://doi.org/10.1109/FG.2018.00019>
- [3] Timothy Bickmore and Justine Cassell. 1999. Small talk and conversational storytelling in embodied conversational interface agents. In *AAAI fall symposium on narrative intelligence*. 87–92.
- [4] Dan Bohus and Eric Horvitz. 2010. Facilitating Multiparty Dialog with Gaze, Gesture, and Speech. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction (Beijing, China) (ICMI-MLMI ’10)*. Association for Computing Machinery, New York, NY, USA, Article 5, 8 pages. <https://doi.org/10.1145/1891903.1891910>
- [5] Justine Cassell, Catherine Pelachaud, Norman Badler, Mark Steedman, Brett Achorn, Tripp Becket, Brett Douville, Scott Prevost, and Matthew Stone. 1994. Animated Conversation: Rule-Based Generation of Facial Expression, Gesture & Spoken Intonation for Multiple Conversational Agents. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH ’94)*. Association for Computing Machinery, New York, NY, USA, 413–420. <https://doi.org/10.1145/192161.192272>
- [6] Justine Cassell, Hannes Högni Vilhjálmsón, and Timothy Bickmore. 2001. BEAT: The Behavior Expression Animation Toolkit. *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2001*, 477–486. <https://doi.org/10.1145/383259.383315>

- [7] Radek Danecek, Michael J. Black, and Timo Bolkart. 2022. EMOCA: Emotion Driven Monocular Face Capture and Animation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. 20311–20322.
- [8] Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. 2020. Learning an Animatable Detailed 3D Face Model from In-The-Wild Images. *CoRR* abs/2012.04012 (2020). arXiv:2012.04012 <https://arxiv.org/abs/2012.04012>
- [9] Epic Gaming. [n. d.]. MetaHuman - Unreal Engine. <https://www.unrealengine.com/en-US/metahuman>. (Accessed on 04/29/2023).
- [10] Patrick Gebhard, Gregor Mehlmann, and Michael Kipp. 2012. Visual SceneMaker—a tool for authoring interactive virtual characters. *Journal on Multimodal User Interfaces* 6 (7 2012), 3–11. Issue 1-2. <https://doi.org/10.1007/s12193-011-0077-1>
- [11] Charamel GmbH. [n. d.]. VuppetMaster® - interaktive 3D Avatare für Websites und Applikationen. <https://vuppetmaster.de/>. (Accessed on 01/06/2023).
- [12] Jonathan Gratch, Anya Okhmatovskaia, Francois Lamothe, Stacy Marsella, Mathieu Morales, Rick Werf, and Louis-Philippe Morency. 2006. Virtual Rapport. 14–27. https://doi.org/10.1007/11821830_2
- [13] Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. 2020. MoGlow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics* 39 (11 2020), Issue 6. <https://doi.org/10.1145/3414685.3417836>
- [14] Lixing Huang, Louis-Philippe Morency, and Jonathan Gratch. 2011. Virtual Rapport 2.0. 68–79. https://doi.org/10.1007/978-3-642-23974-8_8
- [15] Alphabet Inc. [n. d.]. google/mediapipe: Cross-platform, customizable ML solutions for live and streaming media. <https://github.com/google/mediapipe>. (Accessed on 04/26/2023).
- [16] Apple Inc. 2022. Apple AR blendShapes. <https://developer.apple.com/documentation/arkit/arfceanchor/2928251-blendshapes>. Accessed: 2022.
- [17] Kristiina Jokinen and Graham Wilcock. 2014. *Multimodal Open-Domain Conversations with the Nao Robot*. 213–224. https://doi.org/10.1007/978-1-4614-8280-2_19
- [18] Patrik Jonell, Taras Kucherenko, Gustav Eje Henter, and Jonas Beskow. 2020. Let’s Face It: Probabilistic Multi-Modal Interlocutor-Aware Generation of Facial Gestures in Dyadic Settings. In *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents (Virtual Event, Scotland, UK) (IVA ’20)*. Association for Computing Machinery, New York, NY, USA, Article 31, 8 pages. <https://doi.org/10.1145/3383652.3423911>
- [19] Jiwoo Kang, Seongmin Lee, and Sanghoon Lee. 2022. Competitive Learning of Facial Fitting and Synthesis Using UV Energy. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 52, 5 (2022), 2858–2873. <https://doi.org/10.1109/TSMC.2021.3054677>
- [20] Stefan Kopp, Brigitte Krenn, Stacy Marsella, Andrew N Marshall, Catherine Pelachaud, Hannes Pirker, Kristinn R Thórisson, and Hannes Vilhjálmsón. 2006. Towards a common framework for multimodal generation: The behavior markup language. In *Intelligent Virtual Agents: 6th International Conference, IVA 2006, Marina Del Rey, CA, USA, August 21-23, 2006. Proceedings* 6. Springer, 205–217.
- [21] Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexandersson, Iolanda Leite, and Hedvig Kjellström. 2020. Gesticulator: A Framework for Semantically-Aware Speech-Driven Gesture Generation. In *Proceedings of the 2020 International Conference on Multimodal Interaction (Virtual Event, Netherlands) (ICMI ’20)*. Association for Computing Machinery, New York, NY, USA, 242–250. <https://doi.org/10.1145/3382507.3418815>
- [22] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. 2017. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* 36, 6 (2017), 194:1–194:17. <https://doi.org/10.1145/3130800.3130813>
- [23] Nadia Magnenat-Thalmann, Osman Ratib, and Hon Fai Choi. 2013. *3D multiscale physiological human*. Springer.
- [24] Yoichi Matsuyama, Arjun Bhardwaj, Ran Zhao, Oscar Romeo, Sushma Akoju, and Justine Cassell. 2016. Socially-Aware Animated Intelligent Personal Assistant Agent. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, Los Angeles, 224–227. <https://doi.org/10.18653/v1/W16-3628>
- [25] Yoichi Matsuyama, Arjun Bhardwaj, Ran Zhao, Oscar Romero, Sushma Anand Akoju, and Justine Cassell. 2016. Socially-Aware Animated Intelligent Personal Assistant Agent. 224–227. <https://doi.org/10.18653/v1/W16-3628>
- [26] Evonne Ng, Hanbyul Joo, Liwen Hu, Hao Li, Trevor Darrell, Angjoo Kanazawa, and Shiry Ginossar. 2022. Learning to Listen: Modeling Non-Deterministic Dyadic Facial Motion. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022).
- [27] Ana Paiva. 2000. *Affective Interactions: Towards a New Generation of Computer Interfaces*. <https://doi.org/10.1007/10720296>
- [28] Doris Peham, Astrid Bock, Cathrin Schiestl, Eva Huber, Johannes Zimmermann, Dietmar Kratzer, Reiner Dahlbender, Wilfried Biebl, and Cord Benecke. 2015. Facial Affective Behavior in Mental Disorder. *Journal of Nonverbal Behavior* 39 (12 2015), 371–396. Issue 4. <https://doi.org/10.1007/s10919-015-0216-6>
- [29] Catherine Pelachaud. 2009. Modelling Multimodal Expression of Emotion in a Virtual Agent. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences* 364 (12 2009), 3539–48. <https://doi.org/10.1098/rstb.2009.0186>
- [30] Catherine Pelachaud, Carlos Busso, and Dirk Heylen. 2021. *Multimodal Behavior Modeling for Socially Interactive Agents* (1 ed.). Association for Computing Machinery, New York, NY, USA, Chapter 1, 259–310. <https://doi.org/10.1145/3477322.3477331>
- [31] Fabian Ramseyer and Wolfgang Tschacher. 2014. Nonverbal synchrony of head-and-body-movement in psychotherapy: different signals have different associations with outcome. *Frontiers in psychology* 5 (2014), 979.
- [32] Byron Reeves and Clifford Nass. 1996. *The Media Equation: How People Treat Computers, Television, and New Media like Real People and Places*. Cambridge University Press, USA.
- [33] Mark Sagar, Alecia Moser, and Annette Henderson. [n. d.]. Digital People - The Future of CX - Soul Machines. <https://www.soulmachines.com/>. (Accessed on 04/29/2023).
- [34] Soubhik Sanyal, Timo Bolkart, Haiwen Feng, and Michael Black. 2019. Learning to Regress 3D Face Shape and Expression from an Image without 3D Supervision. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 7763–7772.
- [35] Henning Schauenburg and Tilman Grande. 2000. *Operationalisierte Psychodynamische Diagnostik – OPD*. 55–73. https://doi.org/10.1007/978-3-7091-6767-0_4
- [36] Björn Schuller, Ronald Müller, Benedikt Höthker, Anja Höthker, Hitoshi Konosu, and Gerhard Rigoll. 2007. Audiovisual recognition of spontaneous interest within conversations. *Proceedings of the 9th International Conference on Multimodal Interfaces, ICMi’07*, 30–37. <https://doi.org/10.1145/1322192.1322201>
- [37] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. 2020. Face2Face: Real-time Face Capture and Reenactment of RGB Videos. arXiv:2007.14808 [cs.CV]
- [38] Stef van der Struijk, Maryam Sadat Mirzaei, Hung Hsuan Huang, and Toyoaki Nishida. 2018. Facsvatar: An Open Source Modular Framework for Real-Time FACS based Facial Animation. *Proceedings of the 18th International Conference on Intelligent Virtual Agents, IVA 2018*, 159–164. <https://doi.org/10.1145/3267851.3267918>
- [39] Nigel Ward and Wataru Tsukahara. 2000. Tsukahara, W.: Prosodic features which cue back-channel responses in english and japanese. *Journal of Pragmatics* 23, 1177–1207. *Journal of Pragmatics* 32 (07 2000), 1177–1207. [https://doi.org/10.1016/S0378-2166\(99\)00109-5](https://doi.org/10.1016/S0378-2166(99)00109-5)

A DETAILS OF REAL-LIFE THERAPY DATASET

This section provides a comprehensive description of the dataset utilized in this study, which was derived from an extended video corpus collected by Peham et al. [28], Schauenburg and Grande [35]. The original research [35] contains data from 80 women, including 16 healthy controls. In our extended version, a total of 139 video sessions were obtained, from which 134 were chosen for audio and video feature extraction. The remaining five sessions were excluded due to poor recording quality (audio or video). The selected sessions encompass 23 male patients and 113 female patients.

Table 2 presents a detailed breakdown of the dataset, comparing the four therapists who conducted the sessions in terms of the number of sessions, total duration, patient speaking duration, and therapist speaking duration. The segments where patients are actively speaking are defined as the region of interest (ROI) and represent the period during which the therapist is actively listening. By focusing on these ROIs, our study aims to analyze and model the active listening behavior exhibited by therapists during their interactions with patients.

B DATA SEGMENTATION AND DEFINITION OF ROI

To accurately dissect therapist-patient interactions, it was imperative to differentiate between speech and non-speech segments, while optimally segmenting the sessions.

We utilized the speaker diarization tool, pyannote-audio⁷, to segregate speakers, clustering speech segments by their duration into distinct intervals: $S_{\text{no-speech}}$, $S_{\text{backchanneling}}$, $S_{\text{short-speech}}$, and $S_{\text{long-speech}}$. The distinguished speech intervals were:

⁷<https://github.com/pyannote/pyannote-audio>

Table 2: Total number of video hours per therapist sessions. M indicates male and F denotes female.

Therapist ID	No. Sessions	Ttl Duration(h)	Ttl Speech(h)	Patient(h)	Therapist(h)
<i>TherapistA</i> (M)	49 (42F, 7M)	75	54	36	18
<i>TherapistB</i> (F)	51 (42F, 7M)	70	53	43	10
<i>TherapistC</i> (F)	22 (21F, 1M)	22	17	14	3
<i>TherapistD</i> (M)	12 (12F, 4M)	14	12	10	2

$S_{\text{backchanneling}} : [0.5 - 2]\text{s}$

$S_{\text{short-speech}} : [2 - 3]\text{s}$

$S_{\text{long-speech}} : > 3\text{s}$

To extract facial gestures, we employed the EMOCA method [7], which builds upon the FLAME 3DMM model [19]. This method estimates parameters like head-pose, expression, and head-shape. Integrating with mediapipe [15] enabled real-time face detection during inference, with identity-agnostic outputs achieved by omitting shape coefficients.