Listening Style-Aware Dyadic Facial Motion Generation

Gourav Datta[1], Boshi Huang[2,*], Nitesh Sekhar[2], Vivek Yadav[2], Shih-Yao Lin[2], Ayush Jaiswal[2] and Prateek Singhal[2]

[1]University of Southern California, Los Angeles, USA
[2]Amazon Alexa AI, Sunnyvale, USA

Abstract
Modeling dyadic conversation between speaker and listener is the technology that involves spatial/temporal facial motion and language understanding. This has been a key area of interest for building complete conversational artificial intelligence systems. In this work, we propose a generic approach to learn listening styles from multiple listeners and enable facial animations to mimic the listening behavior of conversational avatars. Unlike existing methods which learns one model per person, and cannot generate the listening style of new speakers, our work allows designers to generate new listening styles without requiring any listener data. Furthermore, it is able to generate different listening styles and gives a unique facial expressions and head movements to the listener. Instead of deploying different models for different listeners at runtime, our approach deploys a single model that can generalize to new listeners to generate nonverbal facial responses.

Keywords
Style-aware learning, Spatial-Temporal understanding, Facial motion generation, Multi-modal learning, VQ-VAE

1. Introduction

Conversation with Artificial intelligence is still a long way away from how a regular conversation between two humans looks like. Art of expressing and listening is what makes or breaks a high quality conversation, and in a multi-turn human-to-human conversation, listener behavior is crucial. In most use cases, the avatar acting as listener can interact with the users both verbally as well as nonverbally. During the verbal response, the avatar talks and its facial expression can be synthesized using conversational face and gesture models [1, 2, 3], as well as the audio-driven expression models [4, 5]. The nonverbal response is nondeterministic and hard to model. There have been a few approaches [6, 7] that have attempted to model dyadic conversations. Conversations between people for a similar dialogue can be very different for different listeners just because every person has a unique style (e.g. expressions, head motions) to listen to a conversation. Hence, we need to develop models that can express this back-and-forth of nonverbal facial expressions, eye gaze and head motions during dyadic conversation. Only then, we can enable more natural interactions, increase engagement and build a harmonic user experience for the product that deploys this interactive avatar.

To address this issue, we propose a universal method to learn listening styles from multiple listeners and enable facial animations to mimic listening behavior of conversational avatars. Unlike existing methods that learn one model per person, and cannot generalize to new listening styles, our work allows designers to generate new listening styles without needing any listener data. Additionally, it is able to generate different listening styles and produce unique facial expressions and head motions. We first develop a style representation (embedding) for each listener, and once the style embeddings are available, we sample new listening styles from this embedded space to generate facial expressions and head motions for different listeners. The key highlights of our work are as follows:

- Instead of deploying different models for different listeners at runtime, our approach deploys a single model that can be generalized to new listeners to generate listener facial response;
- Our approach allows users to choose their own listening style or sample the new listening style for conversational avatars, enabling the fine-tuning of the avatars' behavior for specific applications;
- We also propose a novel style transfer aware training techniques to improve the performance of the dyadic facial motion generation pipeline, where the style representation, and the resulting facial motion generation model are trained simultaneously.

STRL'23: Second International Workshop on Spatio-Temporal Reasoning and Learning, 21 August 2023, Macao, S.A.R
**Corresponding author.
gdatta@usc.edu (G. Datta); boshih@amazon.com (B. Huang); seknites@amazon.com (N. Sekhar); ydvivek@amazon.com (V. Yadav); mikeslin@amazon.com (S. Lin); ayujaisw@amazon.com (A. Jaiswal); prtksngh@amazon.com (P. Singhal)
© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3475/paper1.pdf

2. Related Work

Previous works focus on data-driven methods that can predict the 2D motion of a person as a function of the motion of the other person(s) he/she is conversing with [8, 9]. There are some other works that simplify the task of motion generation to predicting head nods [10] or estimating head pose [11]. In contrast, in addition to [12, 13], some recent works [6, 7] capture the natural complexity of interactions by considering the full range of facial expressions and head rotations. While [6] uses a Glow-based model and ingests the full temporal context of listener audio to predict the listener facial motion, [7] proposes a transformer-based predictor and an auto-regressive vector-quantized variational auto-encoder to predict listener facial motion, given past motion and the current speaker audio and facial motion. While [7] does not require the listener audio which can facilitate real-time and synchronized listener motion, it generates a single model per listener ID, and hence, is not generalizable to new/different listeners during runtime. We propose a novel listening style-aware dyadic facial motion generation framework, that can encode the listening style of any person, and generate facial motion corresponding to the encoded style for a particular listener audio and motion. We use the large-scale dyadic facial dataset released by [7] to evaluate our approach.

3. Key Challenges

We identify the main challenges in a robust dyadic facial motion generation pipeline as follows:

1. **Indeterministic Listener Response**: Modeling non-verbal feedback during dyadic interaction is a difficult problem, as listener responses are nondeterministic in nature. This requires the use of probabilistic/generative models, such as GAN, generative model with VQ-VAE, etc., which have been shown to not generalize well to a wide range of diverse, realistic, and indeterministic listening responses during runtime.
2. **Multimodal Problem**: In a dyadic conversational setting, speakers are inherently multimodal, as they communicate both verbally via speech, and nonverbally via face and body motion. This requires the near-perfect alignment of these two modalities to accurately capture the speaker-listener interaction.
3. **Training Complexity & Instability**: The nondeterministic listener motion generation requires the use of generative models such as VQ-VAE, while the multimodal speaker inputs require the use of deep feature extractors, such as cross-modal transformers. This requires the embeddings obtained from the two models to be aligned and fused accurately for realistic facial motion generation, which complicates the training process.

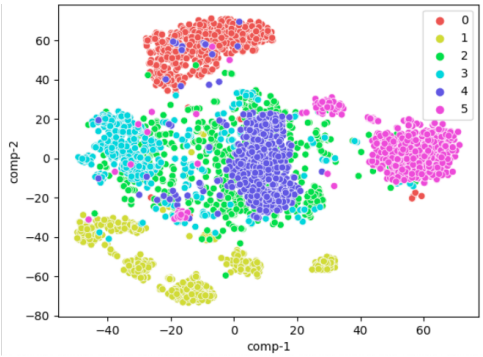

Figure 1: t-SNE projection of listening style. The listening styles can be distinguished using first order moments for each listener in the entire dataset. But there are significant overlaps in the listening styles, which requires a deep learning model to extract the listening style.

4. **One Model Per Style**: Existing works require different models to generate the facial motion corresponding to each style, which is impractical to generalize to new styles during runtime. It is also not feasible to deploy a large number of models to generate a wide range of listening styles in resource-constrained edge devices.
5. **Quantification of Motion Realism**: Quantifying the facial motion realism is a hard problem, as there is no well-defined ground truth (The ground truth listener response from the dataset is unique to a particular style, and might not be the optimal response.) unlike image classification/detection tasks. Naive metrics, such as L2 difference between the predicted and ground truth expression and pose parameters, might not capture realistic listener-speaker interaction in the wild. This motivates the use of other metrics, such as Frechet Inception Distance [14].

4. Methodology

To address these challenges and generate a nonverbal response with facial motion and head movement, we propose a novel framework that extracts two different embeddings: the listening style embedding and motion embedding. Our framework utilizes the popular encoder and decoder architecture and includes a multi-modality predictor that can generate preliminary results and retrieve the final result from pre-trained latent embeddings. By decoding these embeddings, we can generate the avatar's responses that reflect the inherence of the training space.

4.1. Problem Formulation

The aim of dyadic facial generation is to take the speaker's temporal facial motion and audio as input and

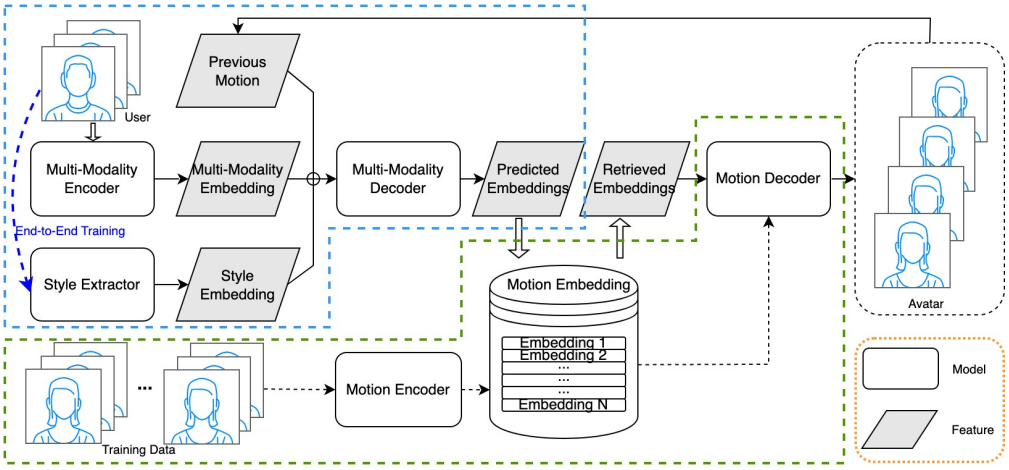

Figure 2: Overview of the proposed listening style-aware framework. The dotted blue area is the main network that predicts the motion embeddings. The model in the dotted green area is the motion encoder-decoder process to generate the latent embedding and may be trained beforehand. The Style Extractor can be trained offline or trained with the prediction model.

generate the nonverbal response. The facial expression and head movement can be extracted with DECA [15] from the speaker's images. We denote the facial expression input as $e(t)$ and head movement as $h(t)$ at time t. The face motion is the combination of the expression and head movement $m(t) = \{e(t), h(t)\}$. The input audio sequence is denoted as $a(t)$ and the listening style is denoted as $s(t)$. $s(t)$ can be trained offline or together with the predictor. If $s(t)$ is trained offline, the $s(t)$ consists of the specific style. If the listener's past n steps are considered, the predictor P can predict the avatar's response at time t' with the input audio, motion and avatar's previous motion:

$$m_a(t') = P(a_i(t'), m_i(t'), m_a(t'-n:t'), s(t')) \quad (1)$$

4.2. Listening Style Analysis

The dataset used in [7] extracts facial features and motion for 72 hours of video from 6 different Youtube channels by using DECA [15], where each channel features a particular host and several interviewees from a variety of backgrounds. Our goal is to extract the listening style (facial expressions and head pose parameters) of the 6 hosts. In order to visualize the listening style of each host, we generate their t-SNE embeddings from the training data, as illustrated in Figure 1. The embeddings indicate that the listening styles can probably be distinguished using first order moments, such as mean of the facial expressions and head pose for each listener ID in the entire dataset. However, we see significant overlaps in some listening styles, such as listener 2 and 4 in Figure 1, and hence, we need to design a model to extract an accurate listener representation style.

4.3. Proposed Model

To surmount the challenges addressed in Section 3, we design a model to generate the natural and realistic nonverbal response to the users (speakers). Therefore, we propose a listening style-aware dyadic facial motion generation framework with a predictor, a style extractor and a motion extractor, as illustrated in Figure 2.

The predictor consists of an encoder and decoder pair, which can be either a Transformer [16], or U-Net [17] based network. The predictor generates the avatar's nonverbal response to the input audio and facial motions.

The style extractor generates the listening style embedding for the predictor. The style embeddings can be fused to the predictor in two ways. We can either extract the listening style embeddings offline, and fuse the embedding corresponding to the desired listening style in the model during runtime. Alternatively, we can train the style extractor model end-to-end with motion prediction pipeline with similar loss, where the style embeddings are fused directly with the multi-modality encoder output. The first approach can generate new listening styles during runtime, and the second approach can better capture the different listening styles with a single model.

The motion extractor is designed to generate the motion latent embedding while reducing the dimensionality. VQ-VAE [18] is firstly used in the image generation model and is being used in [7, 19] for motion generation. The VQ-VAE learns a discrete codebook with multiple vectors to quantize the latent space. Each of the vectors can be looked as an embedding of the input motion.

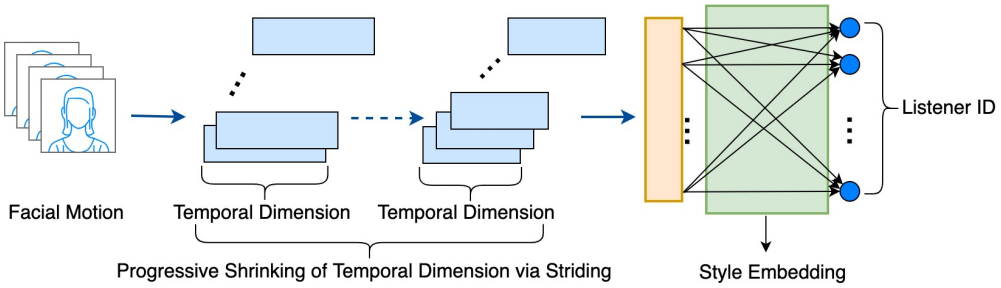

Figure 3: Style extractor network. We use a 6-layer 1-D temporal convolutional neural network model, and each layer has a stride of 2 in the temporal dimension. We can progressively reduce the temporal dimension to 1 in 6 layers. We increase the number of channels from 56 in the input layer to 256 as the style embedding feature, and finally connect to the fully-connected layer for classification.

4.3.1. Style Extractor

We design a 6-layer 1-D temporal convolutional neural network, where each convolutional layer has a stride of 2 in the temporal dimension. Given 64 frames per input, the model progressively reduces the temporal dimension to 1 in 6 layers. We increase the number of channels from 56 (53 expression parameters and 3 head pose parameters) in the input layer to 256, followed by a fully-connected (FC) classifier layer consisting of 6 neurons to predict the listen ID. Our proposed architecture, and the corresponding style embedding output is illustrated in the Figure 3. We employ cross-entropy loss $\mathcal{L}_{\mathcal{E}}$ on the ground truth listener identity distribution $Q(x)$.

$$\mathcal{L}_{\mathcal{E}} = \mathbb{E}_{x\sim p}\left[-log\, Q(x)\right] \tag{2}$$

In addition, we employ the supervised contrastive loss [20, 21] to achieve better performance with limited amount of data:

$$\mathcal{L}_{\mathcal{E}} = \sum_{i\in I} \frac{-1}{|P(i)|} \sum_{p\in P(i))} log \frac{exp(z_i \cdot z_p/\tau)}{\sum_{a\in A(i)} exp(z_i \cdot z_a/\tau)}, \tag{3}$$

where $P(i) = \{p \in A(i) : \widetilde{y}_p = \widetilde{y}_i\}$ is the set of indices of all positives in the multiview batch distinct from i, and $|P(i)|$ is its cardinality. τ is a scalar temperature parameter. z_i, z_p and z_a are the embeddings, which are generated by representation learning, of anchor, positive and negative samples respectively.

We train the contrastive loss with a meta-learning [22, 23] setup, for which we emulate by leaving out a particular listening style, while training with the $N-1$ listening style data, where N is the total number of available listener identities in the dataset.

The contrastive supervision aims to learn the style embedding corresponding to the new listener identity better than the traditional cross-entropy loss, when the number of available samples is limited.

To obtain the listening style embedding, we extract the weights of the fully connected (FC) layer. The subset of the weights connecting each listening style in the output layer is determined to represent the listening style embedding corresponding to that style.

With the style extractor, we can generate diverse listening styles by sampling from the embedding space. For example, we can assume the listening styles to be Gaussian, independently and identically distributed, where the mean and standard deviation of the expression/pose parameters are computed empirically from the style embedding space. Though we have limited number of listening styles, they are quite diverse as can be visualized from the Figure 1, and with more listener data, our approach can be further improved to generate more listening styles.

4.3.2. Motion Predictor

The motion predictor is an encoder-decoder architecture, and we use a Transformer network to better capture the multi-modality feature of the users. The style embedding, previous motion and the multi-modality embedding are concatenated and fed to the decoder to predict the facial motion. When training this model with style extractor end-to-end, we follow the teacher-forcing scheme and train the model with both motion predictor loss $\mathcal{L}_{\mathcal{P}}$ (same as Equ. 2) on codebook index and the style extractor loss $\mathcal{L}_{\mathcal{E}}$ (Equ. 2 and 3) on style ID. The overall loss for the end-to-end training will be:

$$\mathcal{L}_{\mathcal{T}} = \mathcal{L}_{\mathcal{P}} + \mathcal{L}_{\mathcal{E}} \tag{4}$$

At inference step, the predictor predicts multinomial distribution of future facial motion, and by which, we retrieve the closest quantized embeddings from the codebook generated by motion extractor, and send them to the motion decoder to generate the avatar's facial motion.

4.3.3. Motion Extractor

The motion extractor is implemented with VQ-VAE to learn a disrcete codebook $Z = (z_1, \cdots, z_n) \in$

$\mathbb{R}^{256\times n}$to quantize the latent space of the motion. Given an input of sequence with length T of facial motion $M_{1:T} \in \mathbb{R}^{56\times T}$, the encoder will convert it into an embedding $E_{1:\tau} = (e_1, \cdots, e_\tau) \in \mathbb{R}^{256\times\tau}$, where $\tau = \frac{T}{w}$, and w is the temporal window size. Then the embedding will be mapped to the nearest code in the corresponding codebook:

$$z' = arg \min_{z_k^b \in Z^b} \|e - z\| \in \mathbb{R}^{256} \tag{5}$$

And we get the quantized features $Z_{1:\tau} = (z_1, ..., z_\tau) \in \mathbb{R}^{256\times\tau}$.

The decoder takes these quantized features to reconstruct the input motion. The encoder and decoder can be trained simultaneously with the loss function:

$$\begin{aligned}\mathcal{L}_{VQ} = \mathcal{L}_r(M, \widehat{M}) \\ + \|sg[E] - Z\| \\ + \beta \|Z - sg[E]\|,\end{aligned} \tag{6}$$

where $\mathcal{L}_r$ is the MSE loss for the reconstruction, $sg[\cdot]$ is a stop gradient operation [18] to calculate the codebook loss, and $\|Z - sg[E]\|$ is the "commitment" loss with the tradeoff coefficient β [24, 19].

5. Experiments

We conduct extensive experiments to demonstrate the capability of our proposed framework. We implement the framework based on the learning to listen model [7] and evaluate our model on the same dataset. We employ the same metrics to compare with the person agnostic model in [7], which is taken as the baseline. Our method shows significant improvement with both the L2 and the Frechet Inception Distance (FID) [14] metrics.

5.1. Experiment Details

Data We use the dataset released by [7], which contains facial features and head motion for 72 hours of video from 6 different Youtube channels by using DECA [15], where each channel features a particular host and several interviewees from a variety of backgrounds. The corresponding audio melspectrogram features are also generated by audio processing library, librosa [25]. The original videos contain the views of both the host and the guest in a split-screen format. The irrelevant segments have been removed and only the segments that contain the hosts' nonverbal response are kept to extract the pseudo-ground truth.

Motion Extractor Our motion extractor is implemented with VQ-VAE[18]. Similar to the baseline model [7], the motion extractor is composed of 3 convolutional layers of kernel size 5, stride 1, padding 2. Each convolutional layer is followed by a max pooling operation. We pass this bottlenecked sequence through a Transformer of 512 hidden layers, 8 attention heads, 12 attention layers. We train the VQ-VAE on sequences of length 32 for 1000 epochs (1 day on 4 V100 GPUs) with a learning rate of 2.0 with 4,000 warm-up steps. We optimize using Adam with a batch size of 32. Train/val/test split is 70/20/10. We then use the frozen model downstream to quantize the listener inputs to the Predictor.

Multi-Modality Encoder The multi-modality encoder consist of linear layers and a Transformer encoder takes the raw motion representation as input [16]. We feed the audio and the motion independently through a linear layer for each modality to obtain their respective projected embeddings, then send them to the Transformer encoder for cross attention. The Transformer encoder is composed of 1024 hidden layers , 8 attention heads and 12 attention layers. Following the transformer there are 3 convolutional layers of kernel size 5, stride 1, padding 2. Each convolutional layer is followed by a max pooling operation that temporally downsamples the speaker embedding of length 32 to match the size of the listener embedding.

Multi-Modality Decoder The multi-modality decoder is composed of a Transformer decoder with hidden size 200, number of heads 10, and number of layers 5. We concatenate the output of the multi-modality embedding with the listener embedding and previous motion to get a sequence, which serves as input to the multi-modality decoder. During training we take the first 4 indices of the output and discard the remainder. We train the predictor for 1000 epochs (12 hours on 8 GPUs) with a learning rate of 0.01 with 4,000 warm-up steps.

Baseline (L2L)		**Style-Aware (Ours-1)**		**Style-Aware (Ours-2)**		**Style-Aware (Ours-3)**	
FID (*1e3)	L2 Error	FID (*1e3)	L2 Error	FID (*1e3)	L2 Error	FID (*1e3)	L2 Error
	47.04		43.22		42.36		40.28
	47.28		43.43		42.44		40.18
33.5	47.33	24.7	43.45	22.2	42.28	20.3	40.50
	47.21		43.37		42.04		40.53
	47.13		43.20		42.19		40.41

Table 1
Experiment results. Comparison to the baseline[7]. Ours-1 (Train the style representation separately with cross entropy loss) improves the FID by 25% to 24. Ours-2 (Train the style representation separately with contrastive loss) improves the FID by 30% to 22. Ours-3 (Train the style representation model end-to-end) further improves the FID to 20.3.

Evaluation Metrics We employ the L2 and the Frechet Inception Distance (FID) [14] metrics the compare with

100% holdoff		0% holdoff		95% holdoff		85% holdoff		70% holdoff		50% holdoff		20% holdoff	
FID (*1e3)	L2 Error	FID (*1e3)	L2 Error	FID	L2 Error	FID (*1e3)	L2 Error	FID (*1e3)	L2 Error	FID (*1e3)	L2 Error	FID (*1e3)	L2 Error
	54.94		42.36		48.92		47.14		44.75		43.67		42.49
	54.25		42.44		48.75		47.29		44.32		43.60		42.10
39.8	54.83	22.2	42.28	28.4	48.74	25.1	47.19	24.2	44.56	22.9	43.29	22.7	42.56
	54.98		42.04		48.90		47.10		44.68		43.40		42.02
	55.16		42.19		48.79		47.53		47.53		44.41		42.16

Table 2
Exploratory study for different holdoff rate. The FID can be dramatically improved (to 28.4) even by using only 5% of that host's data. With holdoff rate decreasing, the FID is improved gradually.

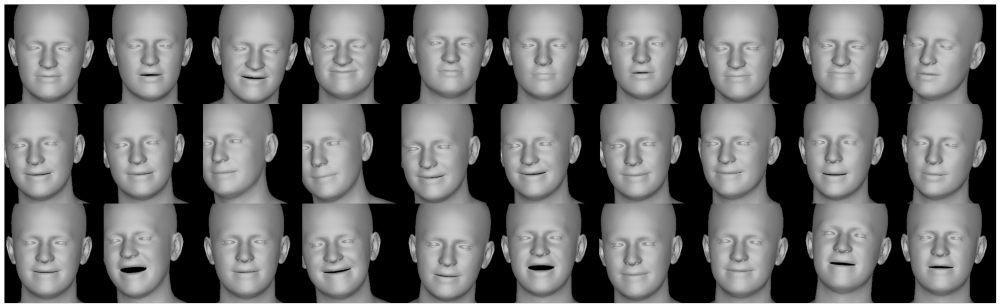

Figure 4: Visual comparison of different listening styles. We sample images from the generated sequences at intervals of 40 frames. Each row of the resulting image sequence represents a distinct listening style, with accompanying facial expression and head pose responding to the same talking video. From the comparison we can see that the proposed method can generate diverse nonverbal response with different listening style.

the baseline model.

- *L2*: L2 distance between ground truth y and the generated facial parameter $\hat{y}$. Here, the ground truth is extracted from listener in the video.

$$L2 = \|y - \hat{y}\| \tag{7}$$

- *Frechet Inception Distance*: FID is a standard metric for assessing the quality of generative models. For two multidimensional Gaussian distributions $\mathcal{N}(\mu, \Sigma)$ and $\mathcal{N}(\mu', \Sigma')$, it is explicitly solvable as:

$$\begin{aligned} d_F(\mathcal{N}(\mu, \Sigma), \mathcal{N}(\mu', \Sigma'))^2 = \|\mu - \mu'\|_2^2 \\ + tr(\Sigma + \Sigma' - 2(\Sigma^{\frac{1}{2}} \cdot \Sigma' \cdot \Sigma^{\frac{1}{2}})^{\frac{1}{2}}), \end{aligned} \tag{8}$$

where $tr(\cdot)$ is the trace of the matrix, i.e. the sum of elements on the main diagonal.

5.2. Experiment Results

We conduct extensive experiment and generate predictions for different listening styles. The metrics comparison is shown in Table 1. Our style-aware model that trains the style representation separately with cross entropy loss (denoted as Ours-1) and with contrastive loss (denoted as Ours-2), improves the FID of the listener-agnostic model by 25% to 24, and 30% to 22 respectively. Training the style representation model end-to-end where it is co-optimized with the main model (denoted as Ours-3) further improves the FID to 20.3. Since our model leads to probabilistic output for realistic listener facial motion, we sample the L2 error of each model 5 times.

We also render the prediction result of different listening style from the ours-2 method to visually compare the result as shown in Figure 4. To facilitate the visual comparison, we sample images from the generated sequences at intervals of 40 frames. Each row of the resulting image sequence represents a distinct listening style, with accompanying facial expression and head pose responding to the same talking video. From the comparison we can see that the proposed method is capable of generating diverse nonverbal response with different listening style.

5.3. Exploratory Study

We also perform exploratory studies to better demonstrate the generalizability of our model with the contrastive loss to learn the listening style, especially when the data is not sufficient. Specifically, we hold out the listener data of one host to train our style extractor model, and extract the mean of the final convolutional layer out-

put of the trained model for the particular listener data in the test set, and use this mean as the listener embedding input to our predictor model. And we train our motion extraction model without the particular listener data. We observed a drop of 80% to 39.8 in the FID by removing the complete data as shown in Table 2 for the listening data of that host. However, we notice that even by using only 5% of that host's data, we were able to recover the FID to 28.4. This was mainly because of the supervised contrastive loss instead of the traditional cross-entropy loss that helps to generalize our model.

6. Conclusion

In this paper, we propose a novel framework and training technique to develop a dyadic facial motion generation pipeline. The goal is to generate the accurate, realistic, and diverse avatar responses for a particular speaker audio and facial motion. We use the open-source, in-the-wild, dyadic conversational dataset released in [7] to evaluate our approach. We conduct extensive experiment to generate the facial response, compare the metrics and visualize the responses. Our method significantly improves the L2 error and FID between the predicted and ground truth facial motion, compared to the listener-agnostic model proposed in [7]. Unlike any other existing approach, our approach can also generate listening response corresponding to a wide range of styles during runtime by sampling in the style embedding space. We will experiment with more modalities and deploy our work to the conversational AI system.

References

[1] Y. Fan, Z. Lin, W. W. Jun Saito, T. Komura, Faceformer: Speech-driven 3d facial animation with transformers, in: CVPR, 2022.

[2] S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, J. Malik, Learning individual styles of conversational gesture, in: CPPR, 2019.

[3] J. Li, D. Kang, W. Pei, X. Zhe, Y. Zhang, Z. He, L. Bao, Audio2gestures: Generating diverse gestures from speech audio with conditional variational autoencoders, in: ICCV, 2021.

[4] L. Chen, G. Cui, C. Liu, Z. Li, Z. Kou, Y. Xu, C. Xu, Talking-head generation with rhythmic head motion, in: ECCV, 2020.

[5] X. Ji, H. Zhou, K. Wang, W. Wu, C. C. Loy, X. Cao, F. Xu, Audio-driven emotional video portraits, in: CVPR, 2021.

[6] P. Jonell, T. Kucherenko, G. E. Henter, J. Beskow, Let's face it: Probabilistic multi-modal interlocutor-aware generation of facial gestures in dyadic settings, in: ACM IVA, 2020.

[7] E. Ng, H. Joo, L. Hu, H. Li, , T. Darrell, A. Kanazawa, S. Ginosar, Learning to listen: Modeling non-deterministic dyadic facial motion, in: CVPR, 2022.

[8] W. Feng, A. Kannan, G. Gkioxari, L. Zitnick, Learn2smile: Learning non-verbal interaction through observation, in: IROS, 2017.

[9] B. Nojavanasghari, Y. Huang, S. Khan, Interactive generative adversarial networks for facial expression generation in dyadic interactions, arXiv preprint arXiv:1801.09092 (2018).

[10] C. Ahuja, S. Ma, L.-P. Morency, Y. Sheikh, To react or not to react: End-to-end visual pose forecasting for personalized avatar during dyadic conversations, in: ICMI, 2019.

[11] D. Greenwood, S. Laycock, I. Matthews, Predicting head pose in dyadic conversation, in: ICIVA, 2017.

[12] G. E. Henter, S. Alexanderson, J. Beskow, Moglow: Probabilistic and controllable motion synthesis using normalising flows, in: TOG, volume 39, 2022, p. 1–14.

[13] D. P. Kingma, P. Dhariwal, Glow: Generative flow with invertible 1x1 convolutions, in: NeurIPS, 2018.

[14] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, in: NeurIPS, 2017.

[15] Y. Feng, H. Feng, M. J. Black, T. Bolkart, Learning an animatable detailed 3D face model from in-the-wild images, in: TOG, 2021.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: NeurIPS, 2017.

[17] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: MICCAI, 2015.

[18] A. van den Oord, O. Vinyals, K. Kavukcuoglu, Neural discrete representation learning, in: NeurIPS, 2017.

[19] H. Yi, H. Liang, Y. Liu, Q. Cao, Y. Wen, T. Bolkart, D. Tao, M. J. Black, Generating holistic 3d human motion from speech, arXiv preprint arXiv:2212.04420 (2022).

[20] S. Chopra, R. Hadsell, Y. LeCun, Learning a similarity metric discriminatively, with application to face verification, in: CVPR, 2005.

[21] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, D. Krishnan, Supervised contrastive learning, in: NeurIPS, 2020.

[22] T. Hospedales, A. Antoniou, P. Micaelli, A. Storkey, Meta-learning in neural networks: A survey, IEEE TPAMI 44 (2022) 5149–5169.

[23] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: ICML, 2017.

[24] X. Chen, K. He, Exploring simple siamese represen-

tation learning, in: CVPR, 2020.
[25] M. Brian, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, O. Nietou, librosa: Audio and music signal analysis in python, in: Proc. Python in Science Conference, 2015.