Let's Face It: Probabilistic Multi-modal Interlocutor-aware Generation of Facial Gestures in Dyadic Settings

Patrik Jonell
KTH Royal Institute of Technology
pjjonell@kth.se

Taras Kucherenko
KTH Royal Institute of Technology
tarask@kth.se

Gustav Eje Henter
KTH Royal Institute of Technology
ghe@kth.se

Jonas Beskow
KTH Royal Institute of Technology
beskow@kth.se

ABSTRACT

To enable more natural face-to-face interactions, conversational agents need to adapt their behavior to their interlocutors. One key aspect of this is generation of appropriate non-verbal behavior for the agent, for example facial gestures, here defined as facial expressions and head movements. Most existing gesture-generating systems do not utilize multi-modal cues from the interlocutor when synthesizing non-verbal behavior. Those that do, typically use deterministic methods that risk producing repetitive and non-vivid motions. In this paper, we introduce a probabilistic method to synthesize interlocutor-aware facial gestures – represented by highly expressive FLAME parameters – in dyadic conversations. Our contributions are: a) a method for feature extraction from multi-party video and speech recordings, resulting in a representation that allows for independent control and manipulation of expression and speech articulation in a 3D avatar; b) an extension to MoGlow, a recent motion-synthesis method based on normalizing flows, to also take multi-modal signals from the interlocutor as input and subsequently output interlocutor-aware facial gestures; and c) a subjective evaluation assessing the use and relative importance of the different modalities in the synthesized output. The results show that the model successfully leverages the input from the interlocutor to generate more appropriate behavior. Videos, data, and code are available at: https://jonepatr.github.io/lets_face_it/.

CCS CONCEPTS

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

ACM Reference Format:
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 *IVA '20: Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents (IVA '20), October 19–23, 2020, Virtual Event, Scotland Uk.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3383652.3423911

IVA '20, October 19–23, 2020, Glasgow, United Kingdom

ACM ISBN 978-1-4503-7586-3/20/09…$15.00
https://doi.org/10.1145/3383652.3423911

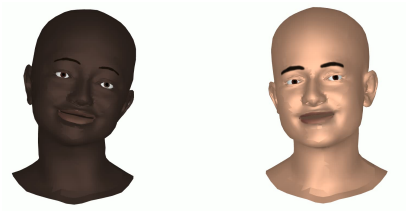

Figure 1: Two avatars in a snapshot from our experiments.

1 INTRODUCTION

Generating appropriate facial gestures (here defined as facial expressions and head movements) for a conversational agent in a dyadic setting is a task as intriguing as it is challenging. Its usefulness in human-agent interaction has been researched extensively [4, 35, 38] and there have been many attempts at realizing its potential in both virtual agents [32, 42] and social robots [45]. It is well known that facial motion is highly correlated with speech, and often contains cues that contribute to or reinforce the spoken message [15]. But facial expressions in a dyadic setting are also strongly affected by the other party. Interpersonal dynamics in face-to-face conversation includes many phenomena that affect the interaction in different ways, such as mimicry – the tendency to adopt poses, facial expressions, mannerisms, and speaking styles of the interlocutor. For example, it has been shown that when a conversational agent just copies the facial expressions of the human interlocutor with some delay it is perceived as more trustworthy [4]. Furthermore, Cassell and Thorisson [9] found that so-called envelope feedback (e.g. gaze, manual beat gestures, and head movements) to be more important for the user than emotional feedback when interacting with conversational agents. As modeling conversational dynamics is difficult to achieve, most non-verbal behavior generation methods only use speech and/or semantic content produced by the agent as inputs to the system [27, 32, 45]. Recently a few systems have been introduced that use non-verbal behaviors from the interlocutor to control non-verbal output from the system [1, 17, 20, 25]. We continue this line of work and present a probabilistic system, based on normalizing flows, for generating facial gestures in dyadic settings. Our system takes in audio from both conversational partners and facial gestures of the interlocutor and generates corresponding appropriate facial gestures for the virtual agent in a given context.

We evaluate this system using segments annotated as containing mimicry from a database of dyadic interactions, these being salient examples of interlocutor-dependent non-verbal behavior.

Our stimulus-generation method allowed manipulating speech articulation independently from the facial gestures, allowing for varying the facial gestures while controlling for the effect of speech context. We find that: (1) Evaluators can distinguish mimicry segments from mismatched segments (from the same interaction but another point in time) and find mimicry segments more appropriate. This also validates that our feature extraction and stimulus generation methods are appropriate for non-verbal behavior. (2) Feeding our model mismatched input segments yields a less appropriate response to the interlocutor, showing that our model leverages the multi-modal signals from the interlocutor to generate more appropriate facial gestures. (3) Removing the interlocutor's facial gestures as input led to less appropriate behavior, while interlocutor speech was not beneficial for facial-gesture generation in our scenario.

In order for researchers to build on top of our work, our extracted database of features and analysis-synthesis code can be found on the project website: https://jonepatr.github.io/lets_face_it.

2 RELATED WORK

2.1 Representing facial communicative signals

While there are many methods for representing facial communicative signals, our scenario and experiments impose the following requirements: Firstly, we require a parameterization that allows encoding facial gestures from video (to be used as inputs and output to the models) in a person-independent way. Secondly, we need to generate an animated 3D avatar, so we require a reliable inversion of the parameterization to render faces that express the perceptually relevant elements. Finally, we need independent control over speech articulation and facial expression, in order to be able to run experiments with out-of-context gestures as in Section 5.

Ekman & Friesen's Facial Action Coding System (FACS) [16] was developed for subject-independent coding of facial expressions for psychology research. It has also been widely used in graphics and machine-learning applications [11, 14, 24], but while FACS is well suited for coding, e.g., emotional expressions, it is less ideal for speech animation. There is also no canonical way of automatically encoding and decoding between video and FACS.

Another commonly used parametrization is facial landmarks, for example the 68 point Multi-PIE scheme, e.g., used in [17]. Facial landmarks often lack resolution and are not fully able to represent facial expressions and emotions [34]. They also lead to subject-specific data and cannot easily be used in generation. MPEG-4 Face Animation Parameters (FAP) are closely related to FACS but were designed to cope with both analysis and synthesis and are, for example, used as output parameters in [14]. There is however a lack of reliable tools for reconstruction/synthesis. Statistically-based 3D analysis/synthesis parameterizations such as 3D morphable models [7] and Active Appearance Models [12] can yield high-quality results, but they typically rely on manual initialization steps that make them expensive to deploy in large-scale multi-talker machine learning settings with many hours worth of data.

FLAME [33] is a new parameterization that represents facial expressions, shapes, and head rotation in a low-dimensional Principal Component Analysis (PCA) parameter space realizable as a 3D mesh. Expression parameters can be automatically extracted from video. Our system uses FLAME parameters as this improves the fidelity of facial gestures. FLAME allows independent control over expression and shape by design. Using techniques described in Section 4.2 it is furthermore possible to independently drive speech articulation and facial expression.

2.2 Gesture generation

Several previous works have demonstrated successful generation of gestures of various kinds. Recent work in speech-driven hand-gesture generation, for example, has primarily been based on deep learning. Hasegawa et al. [22] designed a neural network to map from speech audio to 3D motion sequences. Kucherenko et al. [32] extended this work to learn a better representation of the motion, achieving smoother gestures as a result. Yoon et al. [45] learned a mapping from text to gestures using a recurrent neural network. Speech-driven head-motion and facial gesture generation has been performed using methods such as Variational Autoencoders (VAEs) [31] to predict head pose conditioned on acoustic features [19], Bidirectional Long Short-Term Memory (BLSTM) networks [20, 21, 41], and conditional Generative Adversarial Networks (GANs) [18] as seen in [11, 42]. In another line of work, Karras et al. [28] trained a CNN-based neural network using speech together with a learned emotion representation as input to generate corresponding 3D meshes of faces with impressively little training data.

2.3 Interlocutor-aware gesture generation

Our problem formulation is largely inspired by a recent method to model conversational dynamics for gesture generation [1]. Like in that work, we also model avatar behavior based on both the avatar's own speech and the speech and motion of the interlocutor. One main difference between our work and that paper is that we model a different aspect of non-verbal behavior, namely facial gestures instead of hand gestures. Another important difference is that our method is not deterministic, but probabilistic. Their method is also based on data from motion capture, while our system uses regular videos as input and extracts features from these videos.

One similar work that uses a probabilistic method is DyadGAN [25], which trained a conditional GAN to generate face images based on the interlocutor's facial expressions. However, the work only produced a single image, ignoring temporal aspects. DyadGAN was later extended to generate sequences of interlocutor-aware facial gestures [39]. However, they did not use speech information, nor did they produce output parameters that can control a virtual agent.

Feng et al. [17] presented a system using VAEs to generate facial gestures. However, their system is limited to sequences of facial gestures already existing in the training dataset, while our system is able to generate completely new motions. Furthermore, our system also relies on FLAME parameters for parametrization of the facial features as opposed to facial landmarks, granting several benefits; most importantly, the output parameters can directly generate a high-quality 3D face with corresponding gestures while simultaneously providing independent control over lip-sync and facial shape. Dermouche et al. [14] presented a system similar to Feng et al. but added the conversational state as additional conditional information, and also created a system usable in real time. They encoded the input using LSTMs while outputting FAPs.

2.4 Normalizing flows

In this work we use normalizing flows [40] for probabilistic modeling. This has several advantages over other methods such as VAE or GANs, as detailed in [23]. The specific model we use is adopted from MoGlow [23], which adapted a normalizing-flow method called Glow [30] to the problem of motion generation. We describe the MoGlow method more in detail in Section 3. The method has been successfully applied to gesture generation [3], which inspired us to apply it to our problem as well. However, our system differs from MoGlow as we use several modalities to condition the model, each encoded by a separate neural network, and we apply the model to another task (interlocutor-aware facial-gesture generation). Another difference from both MoGlow and Ajuha et al.'s work [1] is that we start from regular monocular videos, thus not requiring data recorded using specialized motion-capture equipment.

3 SYSTEM ARCHITECTURE

3.1 Problem formulation

We frame the problem of generating interlocutor-aware facial gestures in the following way: given a sequence of speech features of the avatar $S^a = [s^a{}_t]_{t=1:T}$ as well as the interlocutor's facial gestures $F^i = [f^i{}_t]_{t=1:T}$ and speech features $S^i = [s^i{}_t]_{t=1:T}$, the task is to generate a corresponding facial gesture sequence $\hat{F^a} = [\hat{f^a}{}_t]_{t=1:T}$ that the avatar might perform in the conversation.

3.2 Model foundations

The model we utilize to generate motion in this work belongs to the class of probabilistic generative models called *normalizing flows*. Normalizing flows are similar to GANs in that they generate output by drawing samples from a simple *base* or *latent distribution* Z (here a standard normal distribution) and then transform these samples nonlinearly using a neural network g such that the transformed output distribution $X = g(Z)$ matches that of the data. Different from the one-way neural networks in GANs, however, normalizing flows use *invertible* nonlinear transformations, so called *invertible neural networks*, for g. The approach gains power and expressivity by chaining together several simple nonlinear transformations, called *steps* of flow, analogous to the layers in a regular neural network. For more details on normalizing flows please see [40].

The model in this paper is based on a specific normalizing flow transformation g called Glow [30]. This choice allows both fast likelihood computation and efficient sampling from the learned distribution. Our model structure is similar to the MoGlow architecture used for autoregressive generation of pose sequences in locomotion [23] and gesture generation [3]. These papers also show how the nonlinear transformation g, and thus the learned distribution $X = g(Z)$, can be made to depend on conditioning information that affects the motion, including an external control signal. Specifically, MoGlow feeds the conditioning information as an additional input to the regular (one-way) neural networks contained inside each step of flow (see [23]). We will use this control signal to create models of non-verbal behavior that are able to use the interlocutor's speech and facial gestures. Like in MoGlow, we do not use any hierarchical structure in the generator, meaning that $L = 1$ in the language of Kingma et al. [30].

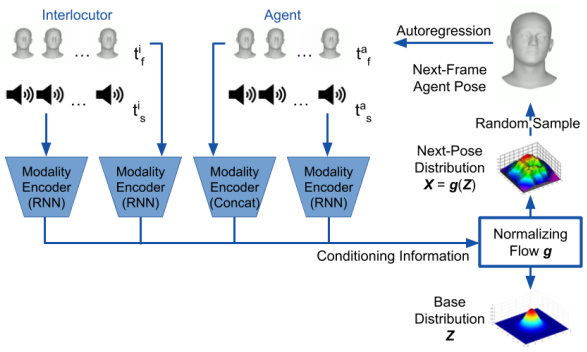

Figure 2: System architecture. While we visualize conversation parties as talking heads in the figure, the facial gesture inputs and outputs of the machine-learning system were FLAME parameters. Similarly, audio inputs were MFCCs and prosodic features, rather than raw waveforms.

3.3 Proposed model overview

Our model generates facial gestures conditioned on the speech of the avatar as well as the speech and the facial features of the interlocutor. A graphical overview of the model is shown in Figure 2. The core of the model is the normalizing flow, which transforms Gaussian driving noise (shown below the model) into a distribution of facial expressions (shown on top of the model). In order to be able to generate smooth facial motion, the model is made autoregressive – it uses the avatar's facial expressions from preceding frames as an extra conditioning to generate the next frame. The generated facial motion should be consistent with the avatar's speech (but not necessarily its semantics) and hence our model is also conditioned on the avatar's speech signal from previous t_s^a time-steps. To enable generating appropriate behavior toward the interlocutor, the speech and facial motion of the interlocutor for the t_s^i and t_f^i time-steps, respectively, are used as additional conditioning for the normalizing flow. The proposed model hence learns to generate a distribution of appropriate facial gestures using multi-modal conditioning.

Since no previous facial expressions are available at test time, the model starts generation with a sequence of zero vectors standing in for the missing facial-gesture inputs.

Like in MoGlow [23] the conditioning information is concatenated with the other inputs to the networks inside the steps of flow, but in our system each modality is encoded by a separate network (and later subjected to an additional transformation which is different for each step), as described in the next subsection.

3.4 Modality encoder

Four different inputs are used in our model to condition the output distribution: the interlocutor's acoustic and facial features, as well as the agent's own acoustic features and previous facial features (as autoregressive input to ensure continuity). How the acoustic and facial features were extracted is described in Section 4.1. Below we describe our modality encoders shown in Figure 2.

We experimented with different neural networks for encoding each modality: Multi-layer Perceptrons (MLPs), Recurrent Neural Networks (RNNs) and 1D-convolution networks (CNNs). We

decided on the final configuration (RNNs) based on an initial hyperparameter search on the validation dataset. However, for the autoregression, the avatar's previous facial features were passed into the normalizing flow model without any processing: simply as a concatenation of t^a_{pf} previous frames. All other modalities were first encoded from input histories of a given time duration (different for different modalities) into fixed-length vectors using separate RNNs, specifically using Gated Recurrent Units (GRUs) [10]. We took both the hidden state and the final output from the GRUs to retain more information. For each step of flow, all modality encodings were concatenated and then passed through a one-layer neural network with a LeakyReLU activation function. This transformation network was different in every step of the flow, resulting in different conditioning vectors in each step. The per-step conditioning information was used to influence the transformation in each step in the same way as in MoGlow.

3.5 Training scheme

We used teacher forcing without annealing or scheduled sampling. This means that the model always received the ground-truth autoregressive input during training instead of samples from the model, since the latter can make models converge on incorrect output [26].

We used the Adam optimizer [29] since it has been used before to train similar systems [3, 23]. We also used learning-rate warm up, as is common for normalizing flows [30]. Different learning-rate schedulers were tested, but did not seem to impact the results.

In order for the model to listen more to the conditioning from the interlocutor we used a special training scheme based on negative learning [37]. The main idea is to not only minimize the loss of the training examples, but also maximize the loss of "wrong", negative examples. There was a 0.1 probability to use a negative sample for each batch. Negative samples are created by shuffling both facial $\boldsymbol{F^i}$ and speech conditioning $\boldsymbol{S^i}$ in the conditioning information for the whole batch, so that each output sequence in the batch now has the conditioning information of a different sample. Temporal consistency was preserved – the mismatched conditioning was still a continuous sequence but from another example. Mathematically, a permutation of elements where no element appears in its original position is known as a *derangement*, but we will refer to such samples with deliberate incorrect conditioning as *mismatched*.

In order to make the model better at distinguishing between appropriate from inappropriate output motion, we want the log-likelihood for mismatched samples to be as small as possible. We therefore switch the sign of the log-likelihood of negative examples. This was done as long as the negative log-likelihood (which we use as the loss in these cases) was positive for those negative examples, an occurrence that became increasingly rare as the loss kept decreasing as the model improved during training.

3.6 Implementation and hyperparameters

Our implementation used the PyTorch-based GitHub repository glow-pytorch[1] as a base, adapted to PyTorch Lightning[2] The hyperparameter search used Optuna [2], which identified the following hyperparameters that we used in our experiments for the proposed model: total conditioning dimensionality = 512, initial learning rate = 10^{-5}, training sequence length = 80. The Glow parameters were K = 16 steps of flow with 128 hidden channels. All other hyperparameters of the final model can be found on the project website. The final model was trained for 15 epochs on a single GPU for approximately 40 hours.

[1]https://github.com/chaiyujin/glow-pytorch
[2]https://github.com/PyTorchLightning/pytorch-lightning

4 DATA

We used the MAHNOB Mimicry Database [6] to train and evaluate the systems in this paper. It contains 11.5 hours of spontaneous dyadic conversations on different topics. The purpose of the corpus was to be able to study dyadic mimicry behavior. The data-gathering used a setup of 15 shutter-level synchronized cameras, two close-talking microphones and one room-capturing microphone. The video streams capturing the faces were gray-scale. 40 participants discussed various subjects over 53 sessions (originally 54 sessions, but one session did not contain data for both participants). The average session length was 13 ± 3.5 minutes. 40 sessions of this dataset have additionally been annotated with mimicry episodes and occasionally their strength. For selecting mimicry segments for the evaluation we used segments annotated for smile, head nod and laughter. For more information, please refer to the original publication [6]. The data was partitioned into an even split of one minute long, randomly-selected segments. We split the dataset in the following way: train 83%, val 10% and test 6.5%. Additionally, one full session was held out completely (the remaining 0.5%).

4.1 Feature extraction

From the videos (one camera angle per person and session) we extracted 2,068,410 image frames at 25 fps. OpenFace [5] was then used in order to extract facial landmarks, which were used to determine bounding boxes for cropping and for the FLAME fitting. Cropped images were fed into RingNet [43] to estimate initial FLAME parameters. The RingNet output together with the facial landmarks were passed into the FLAME fitter in order to determine the final FLAME parameters, which were obtained through two optimizations outlined in [33]. The result was a 100D PCA expression vector, a 12D pose vector with rotations, and a 300D PCA shape vector. From the expression vector we used the 50 first components together with the neck (3D) and jaw (3D) rotations to form our **facial features** (56D). Lastly some temporal smoothing was applied using Savitzky-Golay filtering (window length = 9, polynomial order = 3).

From the audio we extracted 25 MFCCs + 1 log total frame energy (window length = 0.02 s, step size = 0.01 s, nfft = 1024) using python-speech-features [36]. Additionally we extracted prosodic features (pitch, pitch delta, energy, and energy delta) using Praat [8]. The MFCCs and prosodic features were concatenated in order to create the **acoustic features** (30D).

As some of these processing steps were computationally demanding (measurable in CPU+GPU months), the extracted features are publicly available from the project website.

4.2 Stimulus-generation pipeline

A number of processing steps, illustrated in Figure 3, were necessary to generate the video stimuli: First, Voca [13] was used to generate lipsync for all audio within the test segments. Voca takes

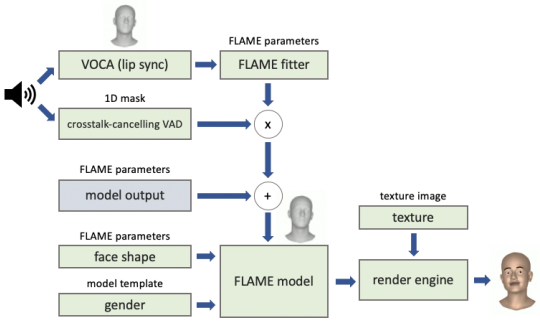

Figure 3: Stimulus generation pipeline, showing how audio is transformed into lip motion and then combined with the model output and rendered.

audio as an input and outputs vertices in the FLAME topology. A template mesh was then fitted to these vertices using the method described in [33] in order to obtain FLAME-parameters for the expression and jaw parameters. A simple energy-based crosstalk Voice Activity Detection (VAD) was implemented to output a mask for canceling crosstalk between the two speakers. This mask was the same length as the number of frames of the FLAME-parameters for the lipsync and was multiplied with each lipsync track. The result was subsequently added together with the model output, resulting in an avatar whose lip-movements are driven by recorded agent speech but whose facial gestures can be generated and manipulated independently. A random gender and a random face shape were sampled in the face-shape parameter space and were, together with the previous output, passed to the FLAME model to obtain 3D vertices. The gender decides which template model the FLAME model will use, and can be generic, female or male. Finally the resulting vertices together with a random texture were passed to the rendering engine, here Pyrender[3].

5 EVALUATION

In this section we describe the subjective experiments we conducted, specifically an ablation study, and the complimentary objective measure used to evaluate our model. We ablated several key components of the model, namely the modalities it used as input and the presence of the special training scheme with negative samples. The specific ablations we considered were: *no-face*: model not conditioned on the interlocutor's facial features; *no-speech*: model not conditioned on the interlocutor's speech features; *no-neg-train*: model trained without the negative samples described in Section 3.5. For each ablation we conducted a separate hyperparameter search on the validation dataset to find the optimal setup and re-trained the models from scratch using the best hyperparameters, to enable the most fair comparison. The exact hyperparameters for these models are provided on the project website.

The ablation study also evaluated how the models perform when they receive mismatched conditioning, to try to understand to what extent the models take the various multi-modal signals into account.

[3]https://github.com/mmatl/pyrender

We call the instances when the avatar's speech was taken from another context "*mismatched* S^a", when the interlocutor's speech was from another context "*mismatched* S^i", and when the interlocutor's facial gestures were from another context "*mismatched* F^i".

5.1 Subjective evaluation setup

Five experiments were carried out on Amazon Mechanical Turk (AMT) to evaluate human perception of the produced facial gestures. The five experiments were designed to answer the following five questions: (1) Can participants discern appropriate facial gestures using our visualization? (2) Does our model take interlocutor input into consideration? (3) What is the importance of the interlocutor's facial features as input? (4) What is the importance of the interlocutor's speech features as input? (5) Does the training scheme with negative samples significantly improve the perceptual quality of output gestures?

5.1.1 Procedure. The procedure of our experiment was similar to that described in [17]. Every participant was first provided instructions and then completed a training phase to familiarize themselves with the task and interface. The training consisted of three items showing the participants what kind of videos they may encounter during the study. Each participant was then asked to evaluate video pairs. In all studies participants compared two videos, each containing two virtual characters interacting with each other (see Figure 1). The participants were always asked to only evaluate the avatar on the right, since it was the only one that was manipulated; the left avatar – the interlocutor – was always the same between both videos, and its movements reflected the same segment of ground-truth motion in the data. The videos were presented side by side and could be replayed separately as many times as desired. For each pair, participants indicated which video they thought best corresponded to the given question and there was also an option to state that they perceived both videos to be equally appropriate. The question we asked was always the same across experiments and similar to that used by Ahuja et al. [1]: "*Which of the two characters on the right side of each video has the most appropriate behavior in response to the character on its left?*"

All subjective tests used a binomial sign test with Bonferroni correction for the five studies. Ties were excluded.

5.1.2 Stimuli. Since the goal was to evaluate facial gestures, audio was removed, but lip-sync, based on the original audio for each character, was retained and was the same between both videos in each evaluated pair. This choice was based on other facial gesture studies such as [17] and on the fact that an informal pre-study (12 participants), found that participants tended to base their judgments on how well the motions matched the semantic content, rather than the interlocutor interaction. We found this to be inappropriate for our study since no explicit linguistic understanding was built into our model. The avatars were placed side by side and facing forward, adjusted such that the 3D avatar would face the viewer when the original talker was facing the other interlocutor in the original interaction. Neck rotation was subtracted from the eyes, giving the impression of the avatar looking straight at the viewer even when turning its head. Head shape, gender and skin color (see Figure 1 for an example) were randomized but kept constant for

each video segment across all experiments. Which conversation party from the original ground-truth interaction that was selected as the interlocutor and placed on the left was based on who spoke the most in that segment, determined by summing the VAD output.

22 video pairs were evaluated in each experiment, except for Experiment 1, where 64 video pairs were evaluated (34 mimicry and 30 non-mimicry segments) and each participant evaluated 10 random pairs of each type. Segments were randomly counterbalanced and (like the original mimicry annotations) varied in duration (3 ± 2 s) from one to eight seconds. All experiments used the same segments, except Experiment 1 which had additional segments as above.

A few randomly-selected examples generated by our method and used for the experiments are available on the project website. Since some of these sequences were jittery, we also provide examples where we "lowered the temperature" of the underlying Gaussian (we set $\sigma^2 < 1$ for $\mathbf{Z}$) [30], which produced smoother motion. We did not apply any smoothing filters to the output in this work.

5.1.3 Participants. All participants were recruited through AMT and were only allowed to participate once in any of the studies. The participants had to have an acceptance rate of at least 98% and completed over 10,000 previous HITs to be eligible for our study. We used attention checks to filter out inattentive participants. For two of the attention checks (one early in the experiment, one close to the end) we added a text telling the participant to report the video as broken. Participants were excluded if they failed any of these attention checks. The other three attention checks comprised pairs presenting the exact same video twice and were placed at the 7th, 10th, and the 15th trial-position for all experimental sessions. Here, an attentive rater should answer "no difference". Participants were excluded if they failed all three of these attention checks.

5.2 Results of subjective evaluation

The results for Experiment 2 (*mismatched*), 3 (*no-face*), 4 (*no-speech*), and 5 (*no-neg-train*) are shown in Figure 4.

5.2.1 Experiment 1: Matched and mismatched ground truth. First we evaluated if our stimulus-generation methods allowed online workers to perceive a difference between the actual facial gestures (*ground truth* condition) and avatar gestures taken from another point in time in the same interaction but with the same person (*mismatched* condition). We recruited 30 participants (14 female, 16 male), all from the USA. Their mean age was 37.4 with a std of 11.1.

We conducted a binomial sign test with Bonferroni correction excluding ties to analyze the responses separately for the two types of stimuli: the mimicry segments and the non-mimicry segments. The ground truth videos were preferred over the Mismatched ones for mimicry segments ($p < 0.001$). There was no statistical significance for the non-mimicry segments (p=1). These results indicate that online workers can indeed distinguish the Mismatched facial gestures from the ground truth, but only in segments where that difference is salient, e.g., if the conversation parties display strong non-verbal interactions such as mimicry. Given this result we concentrated our remaining evaluations on mimicry segments, since they provided for the clearest distinction between appropriate and inappropriate agent behavior. As the non-mimicry segments did not produce a statistical difference they were excluded from remaining studies. Furthermore, since our model required 24 frames (0.96 s) of initialization data, only 22 samples could be used for the remaining experiments.

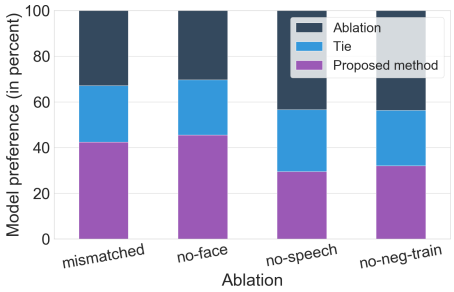

Figure 4: Results from the subjective ablation studies.

5.2.2 Experiment 2: Matched and mismatched proposed model. In the second experiment we evaluated whether the proposed model actually uses the interlocutor's input when generating facial gestures. To this end, we shuffled the conditioning information like before, creating mismatched stimuli where the conditioning information from the interlocutor was always taken from a different sample than the motion used by the interlocutor avatar in the video (but still from the same session and the same person). We compared the proposed model's facial gestures using normal test sequences versus those using mismatched sequences. This use of matched and mismatched samples has the advantage that the quality of the motion is the same across the conditions seen in the videos (since all avatar motion was generated from the same trained model); only the appropriateness of the motion may differ between the two.

We recruited 30 participants (22 male, 8 female). The majority (29) were from the USA. Their mean age was 33.7 with a std of 6.9. The test showed a statistically significant difference between the model output on matched and mismatched test sequences. Specifically, there was a preference towards the matched sequences ($p = 0.032$).

5.2.3 Experiment 3: Ablating facial gestures. Here we compared the proposed model (*proposed* condition) against the ablation where the interlocutor's facial gestures was not available to the model (*no-face* condition). We recruited 30 participants (19 male, 10 female, 1 non-binary), of which 29 were from the USA. The mean age was 37.3 with a std of 9.4. The test showed a statistically significant preference for the proposed model over the *no-face* ablation ($p < 0.001$).

5.2.4 Experiment 4: Ablating speech. In this experiment we compared the proposed model (*proposed* condition) against the ablation where the interlocutor's speech was not available to the model (*no-speech* condition). We recruited 30 participants (16 male, 13 female, 1 non-binary), of which 29 were from the USA. Their mean age was 36.6 with a std of 8.7. The test showed a statistically significant preference for the *no-speech* ablation ($p < 0.001$).

5.2.5 Experiment 5: Negative sample training. In this experiment we compared the proposed model (*proposed* condition) against the same model without any negative samples during training (*no-negative-training* condition).

Table 1: Log-Likelihoods for the proposed model and its ablations on test sequences and mismatched versions thereof.

System	All correct	mismatched S^a	mismatched S^i	mismatched F^i
Proposed	40051±144	40050±144	40050± 144	23522±99436
no-face	38141±240	38141±238	31614±144323	-
no-speech	35545± 67	35544± 68	-	35538± 68
no-neg-train	38698± 92	38698± 93	38699± 92	38654± 97

We recruited 30 participants (17 male, 13 female), of which 28 were from the USA. Their mean age was 38.7 with a std of 12.6. The test showed a statistically significant preference for the model trained without the special training scheme ($p = 0.003$).

5.3 Objective evaluation

It is difficult to evaluate the quality of facial gestures objectively, and it is even harder to objectively evaluate whether or not facial gestures are adapted to the interlocutor. Calculating distance from recorded "ground truth" motion is not meaningful, as a multitude of different gestures can be appropriate even if the conditioning input is fixed. We instead considered the likelihood since normalizing flows enable direct probabilistic inference, letting us calculate the log-likelihood of test data under our model. The test data should have high likelihood only if we model the data distribution well. We evaluated log-likelihood for the proposed model and its ablations for unmodified test sequences as well as mismatched sequences as defined above. The average values along with their standard deviations are given in Table 1. The interpretation of the results is discussed in Section 6.

6 DISCUSSION

The purpose of Experiment 2 (Section 5.2.2) was to see if our method can leverage the multi-modal input to generate more appropriate motion in response to the interlocutor. We found a significant preference for when the model outputs facial gestures relevant to the context, as opposed to a random context, indicating that we successfully generated interlocutor-aware facial gestures. This result is in line with the findings from Experiment 1, where it was shown that evaluators can indeed distinguish – and furthermore prefer – non-verbal behavior which is dependent on the interlocutor over any random (coherent) facial gestures.

Experiments 3 and 4 were designed to assess the relative importance of different interlocutor input modalities. Experiment 3 (Section 5.2.3) considered removing the interlocutor facial information. This made the model perceptually significantly worse. In addition, this *no-face* condition gave likelihoods that were significantly affected by mismatched speech information (Table 1), suggesting that, lacking facial information, the model instead became more attuned to the interlocutor's speech, possibly to the point of overfitting.

If we instead removed the interlocutor speech input (Experiment 4, in Section 5.2.4), the resulting ablation performed significantly better than the proposed model. This suggests that the facial information is the most important for the model, at least in this no-audio evaluation paradigm. It is surprising that the model with facial information alone was better than the one using face and speech together. Speculatively, this might be due to the type of speech features used, and experimenting with less speaker-dependent speech representations would be interesting for future work.

There is an intriguing disparity between the likelihood numbers in Table 1 – where negative training helped models learn to more effectively assign probability mass to motions matching the interlocutor (as opposed to non-matching motion) – and the subjective results from Experiment 5, which found that not using negative samples in the training was perceived significantly better. While negatively-trained models clearly were able to learn to distinguish well between scenarios with matched and mismatched modalities, they do not appear to have leveraged this to generate more appropriate motion in matched setups. However, it is also well known that likelihoods and human ratings are sensitive to different modeling aspects (see, for instance, [44]). Thus higher likelihood does not necessarily mean better perceptual quality, and our findings here are likely another reflection of that fact.

A potential limitation of this work is the fact that we are evaluating multi-modal interactions that contain speech, but without revealing that speech to the evaluators. This was a deliberate choice, as we in a pre-study on mismatched ground-truth motion found that participants otherwise tend to assign an inordinate significance to the linguistic content and how the avatar moves and behaves in relation to that content. Since the presented method does not attempt to model semantics, removing the speech would make it less likely that evaluators assign spurious semantic meaning to the gestures, and instead force them to evaluate the motion in a non-semantic way. It is also consistent with previous evaluation of non-verbal facial gestures, e.g., [17]. Furthermore, we replicated Experiment 2 from Section 5.2.2 with n=30 subjects, but with speech audio present in the video stimuli. We found a statistical difference ($p<0.05$) in agreement with Experiment 2, but the effect was less significant (0.04998), supporting the pre-study finding that the presence of speech with semantic content confounds the evaluation of the non-verbal facial gestures. In general, we believe that the absence of speech audio would be most likely to affect evaluators' assessments of the impact of the speech modalities on the motion, such as the results of Experiment 4. Another limitation is that the evaluated segments, being annotated mimicry episodes, were rather short. In some cases, they may then be considered hard to evaluate.

7 CONCLUSION

We have presented a method for probabilistic and interlocutor-aware facial-gesture generation based on multi-modal inputs. Experiments found that human raters significantly preferred facial gestures generated in response to the interlocutor over mismatched facial gestures that did not take the interlocutor into account. This shows that the proposed approach managed to leverage the multi-modal input to generate better gestures. We evaluated our system on mimicry segments due to their perceptual saliency, but it should be stressed that no information relating specifically to mimicry was used during training. The subjective appropriateness of generated motion decreased significantly when information about the interlocutor's facial gestures was omitted, suggesting that this modality is of major importance to the task.

Future work should investigate the use of other parametrizations of multi-modal signals, especially speech representations, and various ways of incorporating them into the model. It would also be highly interesting to investigate how this method would work in a real-time interaction with a user.

ACKNOWLEDGMENTS

The authors acknowledge the support from the Swedish Foundation for Strategic Research, project EACare under Grant No.: RIT15-0107 and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. (Portions of) the research in this paper uses the MAHNOB MHI Mimicry database collected by Prof. Pantic and her team at Imperial College London, and in part collected in collaboration with Prof. Nijholt and his team of University of Twente, in the scope of MAHNOB project financially supported by the European Research Council under the European Community's 7th Framework Programme (FP7/2007-2013) / ERC Starting Grant agreement No. 203143.

REFERENCES

[1] Chaitanya Ahuja, Shugao Ma, Louis-Philippe Morency, and Yaser Sheikh. 2019. To React or not to React: End-to-End Visual Pose Forecasting for Personalized Avatar during Dyadic Conversations. In *2019 International Conference on Multimodal Interaction*.

[2] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A Next-generation Hyperparameter Optimization Framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[3] Simon Alexanderson, Gustav E. Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-controllable speech-driven gesture synthesis using normalising flows. *Computer Graphics Forum* (2020).

[4] Jeremy N. Bailenson and Nick Yee. 2005. Digital Chameleons: Automatic Assimilation of Nonverbal Gestures in Immersive Virtual Environments. *Psychological Science* (2005). https://doi.org/10.1111/j.1467-9280.2005.01619.x

[5] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. 2018. OpenFace 2.0: Facial Behavior Analysis Toolkit.

[6] Sanjay Bilakhia, Stavros Petridis, Anton Nijholt, and Maja Pantic. 2015. The MAHNOB Mimicry Database: A database of naturalistic human interactions. *Pattern Recognition Letters* (2015).

[7] Volker Blanz and Thomas Vetter. 2003. Face recognition based on fitting a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2003).

[8] Paul Boersma. 2011. Praat: doing phonetics by computer [Computer program]. *http://www. praat. org/* (2011).

[9] Justine Cassell and Kristinn R. Thorisson. 1999. The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. *Applied Artificial Intelligence* (1999).

[10] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *Conference on Empirical Methods in Natural Language Processing* (2014).

[11] Hang Chu, Daiqing Li, and Sanja Fidler. 2018. A face-to-face neural conversation model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[12] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. 2001. Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence* (2001).

[13] Daniel Cudeiro, Timo Bolkart, Cassidy Laidlaw, Anurag Ranjan, and Michael Black. 2019. Capture, Learning, and Synthesis of 3D Speaking Styles. *Computer Vision and Pattern Recognition (CVPR)* (2019).

[14] Soumia Dermouche and Catherine Pelachaud. 2019. Generative Model of Agent's Behaviors in Human-Agent Interaction. In *2019 International Conference on Multimodal Interaction*.

[15] Paul Ekman. 2004. Emotional and conversational nonverbal signals. In *Language, Knowledge, and Representation*. Springer, 39–50.

[16] Paul Ekman and Wallace V. Friesen. 1978. *Facial action coding systems*. Consulting Psychologists Press.

[17] Will Feng, Anitha Kannan, Georgia Gkioxari, and C. Lawrence Zitnick. 2017. Learn2Smile: Learning non-verbal interaction through observation. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*.

[19] David Greenwood, Stephen Laycock, and Iain Matthews. 2017. Predicting head pose from speech with a conditional variational autoencoder. In *Conference of the International Speech Communication Association (Interspeech)*.

[20] David Greenwood, Stephen Laycock, and Iain Matthews. 2017. Predicting head pose in dyadic conversation. In *International Conference on Intelligent Virtual Agents*. Springer.

[21] David Greenwood, Iain Matthews, and Stephen Laycock. 2018. Joint learning of facial expression and head pose from speech. In *Conference of the International Speech Communication Association (Interspeech)*.

[22] Dai Hasegawa, Naoshi Kaneko, Shinichi Shirakawa, Hiroshi Sakuta, and Kazuhiko Sumi. 2018. Evaluation of speech-to-gesture generation using bi-directional LSTM network. In *International Conference on Intelligent Virtual Agents*. ACM.

[23] Gustav E. Henter, Simon Alexanderson, and Jonas Beskow. 2020. MoGlow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Trans. Graph* 39 (2020). https://doi.org/10.1145/3414685.3417836

[24] Hung-Hsuan Huang, Masato Fukuda, and Toyoaki Nishida. 2019. Toward RNN Based Micro Non-verbal Behavior Generation for Virtual Listener Agents. In *International Conference on Human-Computer Interaction*. Springer.

[25] Yuchi Huang and Saad M. Khan. 2017. DyadGAN: Generating facial expressions in dyadic interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*.

[26] Ferenc Huszár. 2015. How (not) to train your generative model: Scheduled sampling, likelihood, adversary? *arXiv preprint arXiv:1511.05101* (2015).

[27] Patrik Jonell, Taras Kucherenko, Erik Ekstedt, and Jonas Beskow. 2019. Learning Non-verbal Behavior for a Social Robot from YouTube Videos. In *ICDL-EpiRob Workshop on Naturalistic Non-Verbal and Affective Human-Robot Interactions*.

[28] Tero Karras, Timo Aila, Samuli Laine, Antti Herva, and Jaakko Lehtinen. 2017. Audio-driven Facial Animation by Joint End-to-end Learning of Pose and Emotion. *ACM Trans. Graph.* (July 2017). https://doi.org/10.1145/3072959.3073658

[29] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.

[30] Diederik P. Kingma and Prafulla Dhariwal. 2018. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*.

[31] Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational Bayes. *The International Conference on Learning Representations (ICLR)* (2014).

[32] Taras Kucherenko, Dai Hasegawa, Gustav E. Henter, Naoshi Kaneko, and Hedvig Kjellström. 2019. Analyzing Input and Output Representations for Speech-Driven Gesture Generation. In *International Conference on Intelligent Virtual Agents*.

[33] 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 Trans. Graph.* (2017).

[34] Caixia Liu, Jaap Ham, Eric Postma, Cees Midden, Bart Joosten, and Martijn Goudbeek. 2013. Representing affective facial expressions for robots and embodied conversational agents by facial landmarks. *International Journal of Social Robotics* (2013).

[35] Pengcheng Luo, Victor Ng-Thow-Hing, and Michael Neff. 2013. An examination of whether people prefer agents whose gestures mimic their own. In *International Workshop on Intelligent Virtual Agents*. Springer.

[36] James Lyons, Darren Yow-Bang Wang, Gianluca, Hanan Shteingart, Erik Mavrinac, Yash Gaurkar, Watcharapol Watcharawisetkul, Sam Birch, Lu Zhihe, Josef Hölzl, et al. 2020. jameslyons/python_speech_features: release v0.6.1. (Jan 2020). https://doi.org/10.5281/zenodo.3607820

[37] Asim Munawar, Phongtharin Vinayavekhin, and Giovanni De Magistris. 2017. Limiting the reconstruction capability of generative neural network using negative learning. In *2017 IEEE 27th International Workshop on Machine Learning for Signal Processing*. IEEE.

[38] Clifford Nass, Jonathan Steuer, and Ellen R. Tauber. 1994. Computers are social actors. In *Proceedings of the SIGCHI conference on Human factors in computing systems*.

[39] Behnaz Nojavanasghari, Yuchi Huang, and Saad Khan. 2018. Interactive generative adversarial networks for facial expression generation in dyadic interactions. *arXiv preprint arXiv:1801.09092* (2018).

[40] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. 2019. Normalizing Flows for Probabilistic Modeling and Inference. *arXiv preprint arXiv:1912.02762* (2019).

[41] Najmeh Sadoughi and Carlos Busso. 2017. Joint learning of speech-driven facial motion with bidirectional long-short term memory. In *International Conference on Intelligent Virtual Agents*. Springer.

[42] Najmeh Sadoughi and Carlos Busso. 2018. Novel realizations of speech-driven head movements with generative adversarial networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '18)*. IEEE.

[43] 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)*.

[44] Lucas Theis, Aäron van den Oord, and Matthias Bethge. 2016. A note on the evaluation of generative models. In *Proceedings of the International Conference on Learning Representations (ICLR'16)*.

[45] Youngwoo Yoon, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2019. Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In *IEEE International Conference on Robotics and Automation*. IEEE.