Integrated Speech and Gesture Synthesis

Siyang Wang
siyangw@kth.se
Div. of Speech, Music and Hearing,
KTH Royal Institute of Technology
Stockholm, Sweden

Simon Alexanderson
simonal@kth.se
Div. of Speech, Music and Hearing,
KTH Royal Institute of Technology
Stockholm, Sweden

Joakim Gustafson
jocke@speech.kth.se
Div. of Speech, Music and Hearing,
KTH Royal Institute of Technology
Stockholm, Sweden

Jonas Beskow
beskow@kth.se
Div. of Speech, Music and Hearing,
KTH Royal Institute of Technology
Stockholm, Sweden

Gustav Eje Henter
ghe@kth.se
Div. of Speech, Music and Hearing,
KTH Royal Institute of Technology
Stockholm, Sweden

Éva Székely
szekely@kth.se
Div. of Speech, Music and Hearing,
KTH Royal Institute of Technology
Stockholm, Sweden

ABSTRACT

Text-to-speech and co-speech gesture synthesis have until now been treated as separate areas by two different research communities, and applications merely stack the two technologies using a simple system-level pipeline. This can lead to modeling inefficiencies and may introduce inconsistencies that limit the achievable naturalness. We propose to instead synthesize the two modalities in a single model, a new problem we call integrated speech and gesture synthesis (ISG). We also propose a set of models modified from state-of-the-art neural speech-synthesis engines to achieve this goal. We evaluate the models in three carefully-designed user studies, two of which evaluate the synthesized speech and gesture in isolation, plus a combined study that evaluates the models like they will be used in real-world applications – speech and gesture presented together. The results show that participants rate one of the proposed integrated synthesis models as being as good as the state-of-the-art pipeline system we compare against, in all three tests. The model is able to achieve this with faster synthesis time and greatly reduced parameter count compared to the pipeline system, illustrating some of the potential benefits of treating speech and gesture synthesis together as a single, unified problem.

CCS CONCEPTS

• **Information systems → Multimedia content creation**.

KEYWORDS

neural networks, speech synthesis, gesture generation

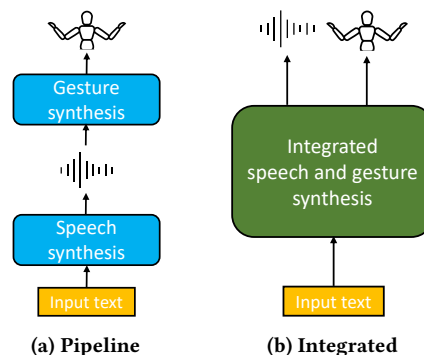**(a) Pipeline**   **(b) Integrated**

Figure 1: Two paradigms for speech and gesture synthesis.

1 INTRODUCTION

Humans make use of both verbal and nonverbal communication to achieve efficient, expressive, and robust face-to-face interaction. Both are fundamentally intertwined, born out of a common representation of the message to be communicated, colored by the situation at hand. Interactions with social robots and embodied conversational agents (ECAs) would benefit from complementing their speech with nonverbal communication like co-gestures [6, 36, 58]. Existing approaches to simultaneous speech and gesture generation have so far simply combined disjunct speech-synthesis systems with gesture-generation components that are trained separately. In this paper, we investigate a fully integrated approach, which we call *integrated speech and gesture synthesis* (ISG).

A common approach to obtaining a speaking and gesturing agent is to stack speech- and gesture generators in a pipeline as shown in Figure 1a. The pipeline generates speech audio from input text using the speech-synthesis component, and then passes that audio to the gesture-generation component to generate matching gestures [3]. Gesture-generation systems can also be driven by text input, which is common for rule-based systems (see 2.2), but for gesticulation and speech to remain synchronized for these agents, it is still necessary for these systems to leverage information from the speech audio, such as word-level timing information (e.g., [9, 61]). This again implies a pipelined approach where speech is synthesized first.

Both speech synthesis from input text and gesture synthesis from input speech audio are well-studied problems on their own.

The state-of-the-art in both areas are data-driven models with deep-learning backbones trained on large datasets. A pipelined text-to-speech then speech-to-gesture approach allows the two components to be trained separately, and potentially on different datasets. The approach also generally benefits from improvement in either component. However, the pipeline approach also has a number of notable drawbacks: First, the output from the speech-synthesis module is usually not of the same quality – most often not even the same voice – as the ground-truth speech audio that the gesture-generation module is trained on. This may degrade the quality of the generated gestures as a result. Second, training two systems separately is inefficient. A typical neural network-based gesture-generation model that takes speech audio as input first needs to extract features from the speech audio, such as the duration, pitch, and intensity of phonemes, in order to generate gestures that match the speech [2, 33]. However, such features are already explicitly or implicitly modeled within speech-synthesis models [46], and forcing the gesture-generation component to extract features from speech audio that already existed inside the speech-synthesis component is suboptimal.

The goal of this paper is to unify speech and gesture generation, in order to bring together the separate research communities of TTS and co-speech gesture synthesis. Our main contributions are:

- We pose and explore the novel problem of building multimodal systems that jointly synthesize speech and gestures in a single, integrated deep-learning architecture.
- We propose two sets of gesture-generation systems based on two representative neural speech-synthesis architectures, namely Tacotron 2 [46] (deterministic, autoregressive) and Glow-TTS [24] (probabilistic, parallel).
- We identify previously-unknown design challenges and trade-offs faced when creating and evaluating integrated systems.
- We evaluate the proposed speech-and-gesture synthesis models in depth against a state-of-the-art pipeline system [3].

Our evaluation considers both synthesized gesture and speech in isolation, along with a third test evaluating time-aligned gesture and speech together. The combined results show that one proposed ISG model achieves same performance as the state-of-the-art pipeline system with 3.5 times fewer parameters and faster synthesis time. The speech and gesture from all evaluated models is included in the supplement. For code and video please see our project page.[1]

2 BACKGROUND

Historically, the synthesis of speech and gesture have been treated as different problems, approached with different goals and different types of data by often disjunct research communities. Lately, however, both fields have moved towards ever more domain-agnostic machine-learning technologies. This section discusses where the two fields are today and highlights recent convergent trends.

2.1 Speech synthesis

State-of-the-art speech synthesis is largely deep learning-based, both for waveform modeling (neural vocoders) following WaveNet [53], and sequence-to-sequence approaches for acoustic modeling (spectrogram generation), first seen in [55] and later popularized by Tacotron [56]. These two trends were brought together in Tacotron 2 [46], which established a new state of the art by separately training a seq-to-seq acoustic model to generate mel spectrograms from text and a neural vocoder to synthesize waveforms from the mels. This is the dominant approach today. The rise of normalizing flows like Glow [26] for neural vocoders [25, 41, 42] and acoustic models [24, 38, 52] has created strong probabilistic TTS models. These can avoid the averaging artifacts seen with deterministic approaches, such as flat intonation and reduced speaker similarity due to over-smoothing. In this paper, we take two leading TTS architectures, one deterministic (Tacotron 2 [46]) and one probabilistic (Glow-TTS [24]), and use them as a starting point for new architectures that generate both speech acoustics and body poses at the same time.

Neural TTS approaches have also seen success in synthesizing convincingly spontaneous-sounding speech [48] with filled pauses [50] and breathing [49]. This ability to synthesize convincing spontaneous speech from text is a key enabler of integrated speech and gesture synthesis, since spontaneous speech generally is accompanied by gestures, while speech read aloud from text (used in the vast majority of TTS training corpora) is not.

2.2 Gesture generation

Co-speech gesture generation has traditionally been dominated by rule-based systems (e.g., [8, 9, 29, 37, 40]; see [54] for a review). The use of machine learning to generate gestures is relatively more recent, and there are also hybrid systems that combine learned and procedural approaches, e.g., by learning when to produce a gesture from a fixed set of pre-animated gesture clips [10, 20, 27, 39].

Among systems that leverage machine-learning for speech-driven gesture generation, one can make a distinction based on the modality used to represent the input speech: either audio, text, or both [34]. Audio-based gesture-generation systems include [13, 14, 18, 19, 31, 35]. Systems of this kind usually generate mainly beat gestures (gestures that align with the rhythm of the speech) and are a natural fit for use in pipeline approaches to speech-and-gesture generation. Text-based systems, on the other hand, are seen as better suited for generating representational and communicative gestures. However, even text-based gesture-generation systems that lack audio as an explicit input, e.g., [20, 61], typically still require word-level timing information to synchronize gestures and speech. This information is not available from text, but only from audio, or from the process that creates it. Finally, in recognition of the fact that both acoustic information (from the speech audio) and semantic information (from a text transcription) are complementary for the task of generating communicative and natural-looking speech-driven gestures, the field is experiencing a rapid shift towards methods that use both audio and text inputs simultaneously [1, 30, 32, 60]. This trend suggests that our proposal for integrated speech-and-gesture synthesis, where the generated gestures may be informed by both text and acoustic properties, is timely and worth pursuing.

2.3 Towards integrated multimodal synthesis

Many embodied agents leverage speech and gesture synthesis components in a pipeline approach. However, the vast majority of these agents use an incoherent setup where the speech synthesis is trained on a different dataset than the gesture generation, and

---

[1] https://swatsw.github.io/isg_icmi21; videos are also in supplemental materials.
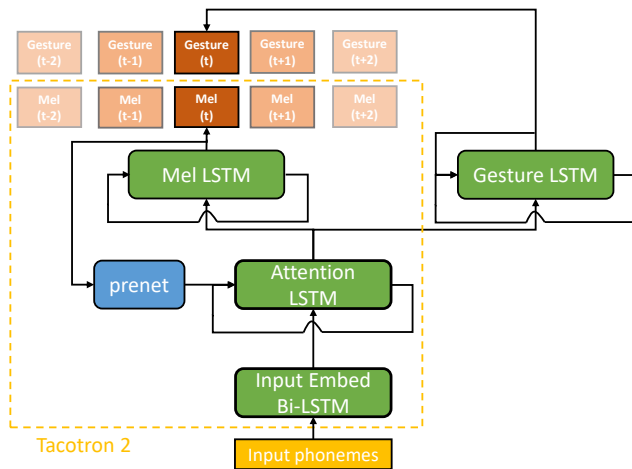
Figure 2: Proposed Tacotron2-based integrated speech and gesture generation model (Tacotron2-ISG). The Tacotron 2 residual conv. postnet that post-processes the mel spectrogram output is also in Tacotron2-ISG but not shown here.

style and speaker identity may differ between components. In fact, the only system we are aware of where both components explicitly were trained on the same dataset is the one in [3], and we consequently use their approach as the baseline pipeline approach for our experiments in Section 5.

In terms of multimodal synthesis beyond speech-and-gesture, the most similar work to ours that we are aware of is DurIAN [62]. Our work differs in three aspects: First, and most important, DurIAN aims to co-generate facial expression and speech, instead of gesture and speech. Gesture generation and facial-expression synthesis are different problems, evident in that they have attracted separate research communities. Synthesizing gesture and speech in a single model is a novel problem that, to our best knowledge, we are the first to study. Second, we use different speech-synthesis frameworks from DurIAN. Third, we use a dataset of spontaneous speech and 3D motion capture from a single speaker, which is very different from the dataset used in DurIAN.

3 INTEGRATED SPEECH AND GESTURE SYNTHESIS

3.1 Problem definition

We define a paired speech-gesture dataset as one that records a human actor gesturing and speaking at the same time and the two modes are time-aligned in the dataset. The speech audio is usually a waveform and the gesture is usually in fixed-frame-rate joint angles or joint positions. Given such a dataset, the problem is to create a machine-learning model that takes text as input and generates both speech audio and gesture. This is an ill-defined problem similar to speech synthesis, due to the fact that the input (text) has much lower information rate than the output (speech and gesture). This means that we have to carefully define the goal for a successful model. While objective measures like mean square error on a held-out set are helpful, what we really care about is how the model is

perceived by human users in an interaction. We therefore evaluate our models in subjective tests, detailed in Sections 5 through 6.

3.2 Proposed models

We develop our ISG models starting from two state-of-the-art TTS models, Tacotron 2 [46] and Glow-TTS [24]. These represent two main approaches to TTS, with Tacotron 2 being auto-regressive and non-probabilistic and Glow-TTS being parallel and probabilistic. The adaptation process detailed below is largely guided by experimentation along with a hypothesis about how to best utilize representations inside TTS frameworks for gesture generation.

3.2.1 Tacotron 2-based models. We adapt Tacotron 2 for ISG by adding a gesture generation sub-network to the architecture and employing different training strategies such as transfer learning, parameter freezing, and an adversarial loss. We focus on two goals, (a) utilizing the intermediate representations in Tacotron 2 to generate gesture and (b) changing as little of the original architecture as possible. We hypothesize that the attention-LSTM layer representation in Tacotron 2 is the most useful for gesture generation because it should correlate with high-level speech-planning. Hence we use this representation as input to the gesture generation sub-network.

We are careful about changing the original speech-synthesis model, because early experiments we conducted found that even minor changes to Tacotron 2 affect the TTS quality. This is especially true for the attention layer: Learning to attend is usually a training bottleneck and attention failures such as skipping and babbling are a weak point of the Tacotron architecture [4, 57]. This issue is especially pronounced when training on small databases [59], and currently-available corpora of spontaneous speech and 3D gesture are often smaller than standard corpora used to train TTS. The monotonic attention mechanism of Glow-TTS [24] is one solution to this problem (among others, e.g., [4, 45, 59, 62]). We tried to add the generated gesture to the input of the prenet which outputs to the attention-LSTM layer, but we found that this small change causes speech synthesis to get much worse. For these reasons, we decide to keep all original Tacotron 2 modules intact (including original hyperparameters [46]), and combine Tacotron 2 with the gesture generation sub-network by only using the attention-LSTM layer output as input to the completely separate gesture-generation module as described above and shown in Figure 2.

We take a fine-tuning approach to train the model. This is mainly because Tacotron 2 and especially its attention LSTM layer is known to be difficult and slow to train from ground up. In fact, it is not uncommon in the speech synthesis community to use transfer learning to train a Tacotron 2 pre-trained on large scale read speech dataset to a smaller speech dataset [48]. We take this approach even further, by first taking a Tacotron 2 model pre-trained on read speech and training it on our speech data only without gestures, and then adding the gesture sub-network for ISG training.

During the ISG training stage, we also experiment with freezing the weights in the speech sub-network, which prevents the possibility that ISG training focuses on improving gesture at the expense of already achieved speech quality. However, this approach takes away supervision from the speech loss which makes it more challenging for the gesture sub-network to generate speech-aligned gesture. Furthermore, the prenet of Tacotron 2 shown in Figure 2,

Siyang Wang, Simon Alexanderson, Joakim Gustafson, Jonas Beskow, Gustav Eje Henter, and Éva Székely

has a random dropout of 0.5 at both train and test time. This means that the attention LSTM layer output varies for the same input text even with weights frozen, making it difficult for the gesture sub-network to regress to the same ground-truth gesture while its input, taken from the output of the attention LSTM layer of Tacotron 2, is constantly changing. This is in contrast to conventional speech to gesture setups, where the input is a finite amount of constant ground-truth speech from the dataset and the model is regressing to the corresponding gesture. On the other hand, this noisy input is very similar to GANs, in which the generator receives random noise input and generates convincing samples by optimizing itself against an adversarial discriminator. Thus, in the model where the speech sub-network is frozen during ISG training, we add a discriminator that takes in both generated speech and gesture as input to encourage the generated gesture to align with generated speech. Inputting both speech and gesture to the discriminator is a practice that has been shown to be effective in encouraging speech-gesture consistency by previous studies [12, 17]. However, an important distinction of ours is that the speech in our model is also generated instead of using ground-truth speech as in those studies.

3.2.2 Glow-based model. We also developed a model building on the recent flow-based TTS system called Glow-TTS [24]. Unlike the autoregressive and LSTM-based architecture of Tacotron 2, Glow-TTS acts in parallel using convolution operators, making it faster to synthesize from on GPUs and avoids stability issues (feedback loops) that are a risk in autoregressive models. We were also interested in the probabilistic aspects of flow-based architectures, since gesture realizations show great variability for a given utterance, and a deterministic summary such as the average gesture cannot capture that diversity.

Glow-TTS has three main components: A transformer-based text encoder f_{enc}, a flow-based convolutional decoder f_{dec}, and a text-to-speech aligner A that time-aligns the input and output, i.e., that associates each mel-spectral frame of speech acoustics with an embedding produced by the encoder. During training, the normalizing flow in the decoder invertibly transforms the mel-spectrogram representation x of a given utterance to a latent representation $z = g(x|c)$, conditioned on the text c. (The invertibility of the transformation g means that we can use the transformation of variables formula to compute the data likelihood. [26]) By letting the latent distribution be elementwise Gaussian and parameterizing it with the alignment A and means and standard deviations $\mu, \sigma = f_{enc}(c)$ returned by the decoder, one obtains a probabilistic model $P_X(x|c)$ of the original speech given the text. The model can then be optimized by gradient-based methods using a combination of maximum-likelihood estimation and Viterbi decoding. During synthesis, $f_{enc}(c)$ and A are used to predict speech-sound durations and then sample a sequence of conditional random latents from $P_Z(z|c)$ that conforms to these durations. The decoder then transforms these sampled latents into mel-spectral frames, thus obtaining a sample from the distribution $P_X(x|c)$. The resulting mel-spectrogram x is passed to a vocoder (HiFi-GAN [28]) to generate a speech waveform.

In this work, we let the decoder model the multimodal distribution $P_X(x_a, x_m|c)$, where the subscripts a and m denote *audio* and *motion*, by simply extracting audio and motion features at the same

time instances (same frame rate) and concatenating them into a unified vector for each frame. Glow-TTS being a normalizing flow model, its layer-wise representation is entangled thus making it difficult to use any one layer as input for generating gesture as was done for Tacotron2-ISG.

4 DATA

4.1 Training corpus

For the experiments we used the recordings from the Trinity Speech-Gesture Dataset [11] as processed by [33]. The dataset comprises 25 impromptu monologues by a male speaker of Irish English, on average 10.6 minutes long, from a multi-camera motion-capture studio. The actor speaks in a colloquial style, spontaneously and without interruption on topics such as hobbies and interests. During the monologues, he addresses a person seated behind the cameras who is giving visual, but no verbal feedback.

To create a corpus suitable for speech synthesis, the audio recordings were automatically segmented into breath groups with an automatic breath detection method [51]. Words were transcribed using Google Cloud Speech-to-Text API and subsequently manually corrected to ensure that all words were accurate. All speech events were transcribed using ARPABET phones, no new characters were introduced outside of the standard. In order to maximize the utterance length in the corpora and to enable insertion of inhalation breaths in the TTS, we used a data augmentation method called *breathgroup bigrams*. This method essentially consists of segmenting a speech corpus into stretches of speech delineated by breath events, and then combining these breath groups in an overlapping fashion to form utterances no longer than 11 seconds [49]. This method also makes it possible to leverage the continuous nature of the recordings and learn contextual information beyond respiratory cycles. The minor misalignment between motion and speech in the original dataset was manually corrected. The motion is represented by exponential map of joint rotations [16]. We only used the upper body data and removed the fingers (but not hand orientation) due to low capture accuracy there. For visualization, we instead used fixed, lightly-cupped hands on the avatar, similar to [3].

4.2 Text inputs for the evaluation

Selecting the input text for evaluating spontaneous speech synthesis is not straightforward, because the training data does not conform to the conventions of written language and lacks a clear sentence structure [47]. In this work, the evaluation uses text prompts that were semi-automatically generated by the medium-size pre-trained GPT-2 model [43] (355 million parameters) fine-tuned on the TTS corpus. Since there are no sentence boundaries in the spontaneous speech corpus, commas indicating pauses and periods were used as breath tokens and end-of-utterance markers. The last breath token was replaced by an end-of-utterance marker. When generating the prompts, common first-person phrases were used as prefixes to seed the GPT-2 samples, and these prefixes were included in the generated prompt. The generated texts underwent a manual selection process to identify 17 semantically coherent utterances mostly between 25 and 50 tokens long. We find that longer sentences are more suitable for evaluating ISG because they differentiate models more than shorter sentences do, since models have to generate more

gesture strokes while being consistent. Using GPT-2-generated input sentences means that we do not have access to corresponding ground-truth speech and gesture. This makes it difficult to assess the gap between the synthesis and the ground truth, but addressing this question is not the main purpose of this study. In addition, using generated sentences allows us to evaluate model performance "in the wild" to some degree.

5 EVALUATION

5.1 Model and training configurations

5.1.1 Baseline: Pipeline with Tacotron 2 and StyleGestures. We compare the models against a state-of-the-art pipeline system [3] which uses Tacotron 2 for speech synthesis and StyleGestures [2] for gesture generation. The only changes we made are using WaveGlow [42] for vocoding, as opposed to the Griffin-Lim algorithm used in [3], and only training on upper-body data. We used the publicly available, official implementation of StyleGestures[2], using the recommended hyperparameters with 16 flow-steps and 512 channels in the LSTM layers. The model was trained for 80k iterations. We initialized the autoregressive context with a static mean pose and padded the test audio with 1 s of silence at the end to account for the future speech data input the gesture-generation model requires.

5.1.2 Tacotron2-ISG. We use an open-source Tacotron 2 repository [3] and the pre-trained model that it comes with. We also use the pre-trained WaveGlow model in that repository for vocoding. Unless stated otherwise, the hyper-parameters used in both speech-only training and ISG training are the same as the repository defaults. We then add the gesture sub-network consisting of 4 LSTM layers, each with 512 nodes, the input of which is the attention-LSTM layer output of Tacotron 2 as shown in Figure 2 together with the prior frame gesture output for autoregressive learning. The resulting model, which we call *Tacotron2-ISG*, is then trained for integrated speech-gesture generation in two ways, (a) with both speech and gesture sub-networks trained simultaneously, which we call Co-training Tacotron2-ISG or *CT-Tacotron2-ISG*, (b) the speech sub-network weights frozen after adding the gesture sub-network, and training only gesture sub-network, which we call Separate-Training Tacotron2-ISG or *ST-Tacotron2-ISG*. In all cases the training loss is mean squared error (MSE), but as mentioned in Section 3.2.1, the ST-Tacotron2-ISG model (in which the speech sub-network is frozen) also has an added GAN loss on the combined speech-gesture output, so that the speech sub-network can provide additional supervision despite being frozen. The GAN loss is in its original form [15] and is weighted by 0.05, as we found that too strong a GAN loss results in bad gestures. We use a discriminator consisting of 2 LSTM layers, each with 1024 nodes. We apply scheduled sampling [5] when training the gesture sub-network, where the teacher forcing probability is 1 for the first 5 epochs, then linearly drops to 0.2 over 40 epochs, remaining at that value for the rest of the training. We do not use scheduled sampling to train the speech sub-network because Tacotron 2 already works well with full teacher forcing. The gesture-generation sub-network is also autoregressive as it takes the prior frame (pose) as input, in addition to the output

Table 1: Model parameter counts and average synthesis time with 95% confidence intervals.

System	Param. count	Synth. time
Pipeline [3], comprising 2 sub-systems:	137.53M	5.08±0.49 s
TTS: Tacotron 2 [46]	28.19M	1.56±0.15 s
gesture: StyleGestures [2]	109.34M	3.52±0.34 s
Tacotron2-ISG (ours)	38.83M	1.49±0.13 s
GlowTTS-ISG (ours)	28.95M	1.64±0.12 s

of the attention-LSTM in the speech sub-network. We find that the gesture sub-network works best running at 20 fps. However, the attention-LSTM layer of the speech sub-network (Tacotron 2) runs at roughly 80 fps. Thus, the gesture sub-network takes output from the attention-LSTM layer every 4 frames to match its own 20 fps rate. In order to match the gesture smoothness of the other models, we also apply 1-D Gaussian filters to the generated gesture in sliding windows with stride 1 and window size 3.

5.1.3 GlowTTS-ISG. For the modified Glow-TTS model we used a temporal resolution of 60 fps for both the audio and motion features, and re-trained the HiFi-GAN vocoder [28] to generate audio at this frame rate. For model training, we used the same hyper-parameters as [24], except for the number of blocks in the one-by-one (depthwise) convolution in each flow layer, where we used 25 blocks of 10 features instead of the original 40 blocks of 4 features. This results in slightly longer training and synthesis time but ensures a more expressive model through better channel mixing. We followed the recommended procedure of adding blank spaces between each text-token and trained the model for 275k iterations. We then generated evaluation samples with a temperature of 0.7 and length scale of 0.9, providing a good mix between output variation and naturalness. We call this model *GlowTTS-ISG*.

5.2 Model size and synthesis time

Table 1 reports the number of parameters of the compared models. Both ISG models have comparable parameter count, and have at least 3.5 times fewer parameters than the pipeline system. If either ISG model obtains just same level of perceptual evaluation results as the pipeline system, we can say that the ISG model is more parameter-efficient. Table 1 also shows that the pipeline model has the longest synthesis time, both as a result of its sequential nature (i.e., the gesture sub-system cannot start running before the TTS completes) and due to the computation-heavy gesture sub-system StyleGestures.[4] Tacotron2-ISG is faster since it eliminates the latency between TTS and gesture synthesis and has a more efficient gesture module. GlowTTS-ISG has comparable synthesis time to Tacotron2-ISG on the tested inputs, but is expected be faster on longer inputs due to its non-autoregressive design. The time used by the vocoder (WaveGlow or HiFi-GAN) is not included in the measurements, since either system can be used with either vocoder. To ensure that the difference in synthesis times reflects

---

[2]https://github.com/simonalexanderson/StyleGestures
[3]https://github.com/NVIDIA/tacotron2

[4]The results we report here are based on a modification of the official implementation from https://github.com/simonalexanderson/StyleGestures, to cache the inverse matrix computations in the flow. This sped up generation time by approximately a factor five.

For each video, please rate:

How appropriate is the gesture for the speech?

Volume ●────────

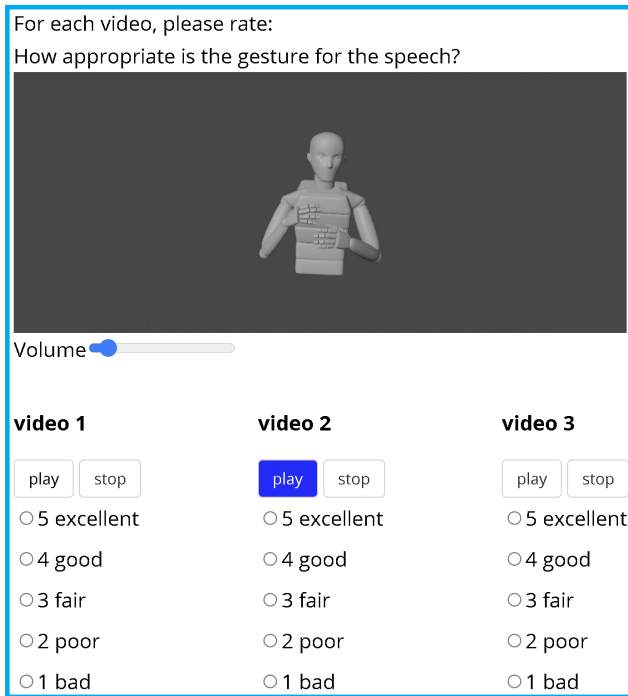| video 1	video 2	video 3
play stop	play stop	play stop
○ 5 excellent	○ 5 excellent	○ 5 excellent
○ 4 good	○ 4 good	○ 4 good
○ 3 fair	○ 3 fair	○ 3 fair
○ 2 poor	○ 2 poor	○ 2 poor
○ 1 bad	○ 1 bad	○ 1 bad

Figure 3: Rendered video and interface for evaluation.

different complexity of the models, and not length differences in the synthesized utterances, we calculated the mean utterance duration for the evaluated systems, finding them to be roughly the same: 8.33 s (Pipeline), 8.26 s (Tacotron2-ISG), and 9.76 s (GlowTTS-ISG).

5.3 Perceptual evaluation method

We evaluate the proposed models in three separate tests: speech-and-gesture, gesture-only, and speech-only. Assessing how models perform in each modality is necessary as users may be biased towards one modality when rating speech-and-gesture generation, as demonstrated in a recent study focused on gesture-generation evaluation [33]. It showed that mismatched ground-truth gesture, i.e., ground-truth gesture taken from a different speech segment, receives better scores than model-generated gestures on this dataset, likely because the motion itself, albeit unrelated to the speech, is more natural than model-generated motion. Evaluating both modalities separately also allows us to understand how each modality contributes to the overall success of a model, and whether or not a model is biased towards either modality, for example generating better speech at the expense of worse gesture.

We render the generated gesture on a skinned avatar [33] with time-aligned speech at 20 fps. We choose 20 fps because it is the lowest frame rate among the models. All models are trained at frame rates that they perform best at (for both speech and gesture). A screenshot of the resulting rendered video is shown in Figure 3.

We use a MUSHRA-like [21] (MUltiple Stimuli with Hidden Reference and Anchor) interface commonly used for subjective evaluation of speech-synthesis [44], but here adapted for video interfaces, since such setups have been found to work well for

evaluating head motion and hand gestures [7, 23, 33]. On a single test page, participants are presented with videos of generated gesture-speech from all evaluated models on the same input text sentence. They can play in any order for any number of times, and rate each video based on a test question, on a standard MOS scale [22]. The order of the test pages and the order of videos on each page is independently randomized for each user. This setup is used for both video-based tests (speech-and-gesture and gesture-only). The speech test uses a similar MUSHRA-like interface. A screenshot of the video-evaluation interface is shown in Figure 3. We recruit three separate groups of native English speakers for the three tests on the Prolific crowdsourcing platform.

6 PERCEPTUAL EVALUATION RESULTS

6.1 Speech-and-gesture evaluation

We asked 23 users to rate "How appropriate is the gesture for the speech?" for each model-generated speech-gesture pair. This question is taken from [33] and is intended to assess the coherence between gesture and speech in general, including synchrony and meaningfulness. GlowTTS-ISG is not evaluated in this test, nor in the speech-only test, since the generated speech quality does not approach the intelligibility standards necessary for meaningful perceptual evaluation. The results are shown in Figure 4. ST-Tacotron2-ISG obtains the highest MOS at 3.35 while the pipeline system scores 3.31, however the difference is not statistically significant.[5] We note that ST-Tacotron2-ISG achieves comparable performance to the pipeline despite being much more parameter efficient in its gesture generation. CT-Tacotron2-ISG which updates both the speech and gesture sub-networks simultaneously scores lower than other models (statistically significant).

6.2 Gesture-only evaluation

In this evaluation we ask participants to rate "How human-like is the gesture?" for each model-generated gesture. This question is also taken from [33]. It assesses how closely the generated gesture

---

[5]Unless otherwise noted, statistical significance is tested using pairwise t-tests at p=0.05.
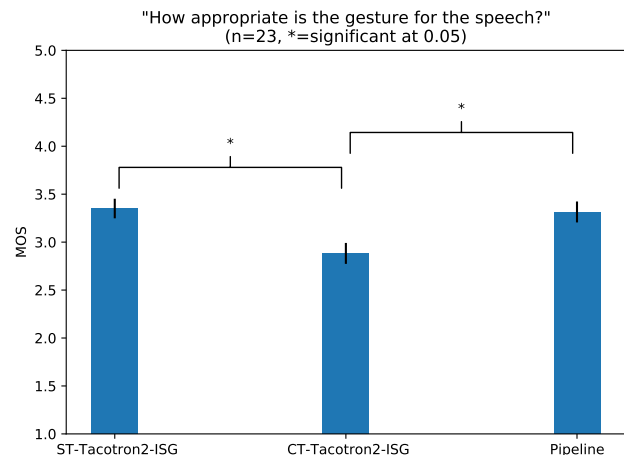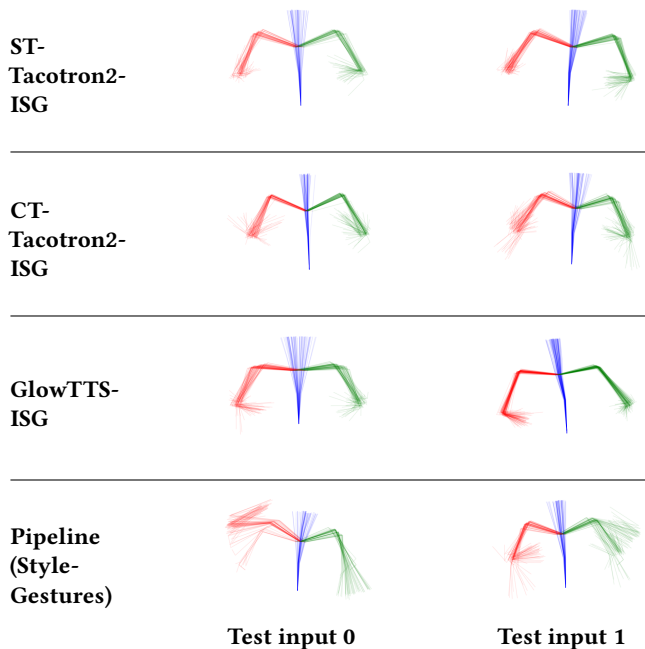

Figure 4: Speech-and-gesture evaluation result.

ST-Tacotron2-ISG

CT-Tacotron2-ISG

GlowTTS-ISG

Pipeline (Style-Gestures)

Test input 0   **Test input 1**

Figure 5: Visualizations of generated gesture space. Colors distinguish right arm (red), left arm (green), and torso (blue).

Figure 6: Gesture-only evaluation result.

motion resembles ground-truth motion. The videos used in this test are the same as the ones in the speech-and-gesture evaluation with GlowTTS-ISG videos added, but with audio turned off in order to remove the effect of the speech when assessing motion.

Figure 6 shows the result of the gesture-only evaluation. The pipeline system (StyleGestures [2]) scores highest at 3.53 while ST-Tacotron2-ISG scores second best at 3.44, however the difference between the two is not statistically significant. GlowTTS-ISG and CT-Tacotron2-ISG score lowest. Their difference is not statistically significant. StyleGestures is significantly better than these two models while ST-Tacotron2-ISG is only significantly better than CT-Tacotron2-ISG. We find StyleGestures to be very dynamic and having more detailed motion, such as subtle head bobbing when talking fast. This is consistent with it scoring highest in this evaluation. On the other hand, ST-Tacotron2-ISG is comparable to StyleGestures while having 3.5 times less parameters.

Figure 5 visualizes two evaluated gesture sequences generated by each of the 4 models for the same two inputs. All models learn to generate plausible gesture shapes. StyleGestures (row 4 in the figure) has the greatest range and variation, which is consistent with it receiving the highest rating in the gesture-only evaluation.

6.3 Speech-only evaluation

The TTS community currently largely relies on questions similar to "How natural does the speech sound?" to evaluate synthesized speech. However, some of the models we evaluate have speech errors such as skipped words, an aspect that we also want listeners to evaluate. We thus combine naturalness and intelligibility evaluation and ask listeners to "Please rate the synthesized speech audios based on a combination of: a) whether or not you can clearly hear
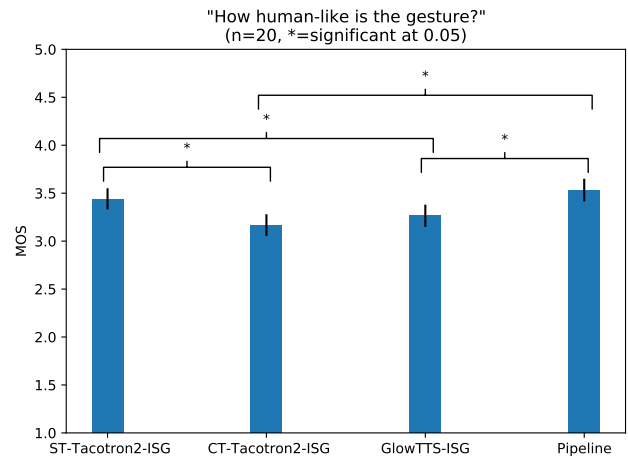
each word, and b) how natural they sound." The input text is shown to users in this test, which is not the case for the other two tests.

The models compared in this test are different than the other two tests. We test three versions of Tacotron 2 to understand how ISG training affects the generated speech. The three versions are, (a) full ISG from scratch (no speech-only pre-training), (b) speech-only training, and (c) ISG fine-tuning after speech-only training. All three are trained for the same number of iterations. The speech-only model is the same as the speech sub-network of ST-Tacotron2-ISG and the pipeline system, which is just Tacotron 2 by itself. GlowTTS-ISG is not evaluated in this test due to its low speech quality.

The results are shown in Figure 7. ISG fine-tuning on top of speech-only training obtains the highest score at 3.62, better than both other training setups. This could be due to increased training time, since the ISG system fine-tuned on top of the speech-only model, but even if that is the case, it still establishes that ISG can sound equally good as speech-only systems. Full ISG training from scratch obtains the lowest score at 2.49, significantly worse than the second best speech-only training which obtains 3.49. We think this is because the database size is smaller than what Tacotron 2 typically needs to create high-quality TTS, showing the benefits of a transfer-learning approach that leverages unimodal data for ISG.

7 DISCUSSION

ST-Tacotron2-ISG scores higher than CT-Tacotron2-ISG in both speech-and-gesture and gesture-only evaluations. The biggest difference between the two models is that CT-Tacotron2-ISG trains both speech and gesture sub-networks together while ST-Tacotron2-ISG freezes the speech sub-network during gesture training. This suggests that full ISG training, where both sub-networks are optimized simultaneously, may result in worse gesture as the overall model focuses on improving speech. The speech loss could be given a reduced weight to balance the two modalities. Moreover, weighting in general is a potential way to improve overall synthesis quality by changing which modality the model focuses on during a certain training phase. The speech-only evaluation reveals that transfer learning from larger, unimodal databases may be used to boost
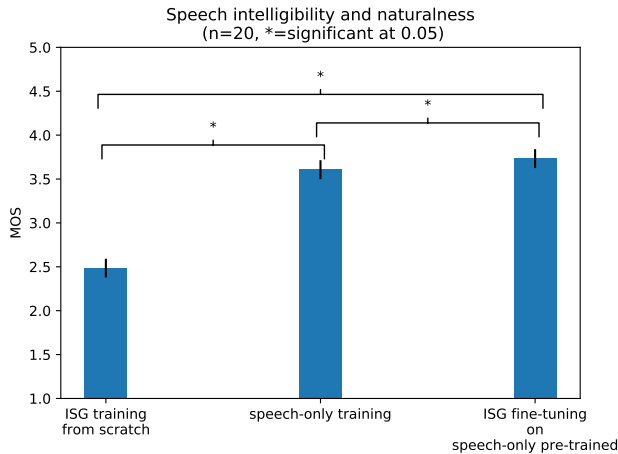
Figure 7: Speech evaluation result.

ISG quality and convergence time in both modalities. This is particularly appealing for ISG approaches, since speech data is much more widely available than aligned speech and 3D motion-capture gesture material.

One aspect of the proposed Tacotron2-ISG models we are particularly interested in is how the Tacotron 2 attention-layer informs gesture generation. To probe this, we trained a similar model to Tacotron2-ISG in which the gesture sub-network takes the generated mel-spectrogram from Tacotron 2 as input, instead of the output from Tacotron 2 attention-layer. We found that the model generated less articulated gestures than Tacotron2-ISG.[6] This suggests that the Tacotron 2 attention layer provides features that facilitate gesture generation and that are not exposed by a pipeline approach. However, other speech synthesis-frameworks may process information differently, and may be even more suited for an ISG approach.

We also observed that the proposed Tacotron2-ISG model is able to reproduce common speech-gesture patterns such as a subtle shrug and symmetrical hand gestures when saying "I don't know". While the model itself does not have semantic input, the dataset contains several occurrences of "I don't know" in different contexts, and it is possible that the gesture sub-network has learned to associate the attention-layer representation of that phrase with the shrugging gestures in the database.

Furthermore, the poor synthetic speech generated by GlowTTS-ISG does not imply that a probabilistic generative model cannot achieve good ISG. The issues may at least partially be explained by database size, since normalizing flows endeavor to learn the dual-modal data distribution, whereas minimizing the MSE only requires being able to predict its mean. However, the strong performance of unimodal flow-based systems such as StyleGestures show that these models still have plenty of potential for ISG.

8 LIMITATIONS

As presented here, integrated speech and gesture generation relies on a database of text, audio, and 3D motion in parallel. Such

databases are smaller and less common than databases containing only a subset of these modalities. Furthermore, we only investigated two TTS models; other TTS models may also be suitable for the task. Lastly, evaluation of speech and gesture remains challenging for the research community in general, and innovations in evaluation approaches could reveal additional nuances of the tested models.

9 CONCLUSIONS

We introduce integrated speech and gesture generation (ISG), where both modalities are generated jointly in a single architecture, as a new research problem that brings together TTS and gesture generation. We propose several models for ISG and evaluate them in a set of carefully designed user studies, on each modality separately and on both modalities combined. Taken together, the results from all three studies demonstrate that one of the proposed ISG models (ST-Tacotron2-ISG) performs comparably to the current state-of-the-art pipeline system, while being faster and having much fewer parameters. Our findings, and the challenges we identified along the way, suggest that ISG is a promising and largely unexplored topic which deserves attention from synthesis researchers across communities.

ACKNOWLEDGMENTS

REFERENCES

[1] Chaitanya Ahuja, Dong Won Lee, Ryo Ishii, and Louis-Philippe Morency. 2020. No Gestures Left Behind: Learning Relationships between Spoken Language and Freeform Gestures. In *Proc. EMNLP*. 1884–1895.

[2] Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-Controllable Speech-Driven Gesture Synthesis Using Normalising Flows. *Computer Graphics Forum* 39, 2 (2020), 487–496.

[3] Simon Alexanderson, Éva Székely, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Generating coherent spontaneous speech and gesture from text. In *Proc. IVA*. 1–3.

[4] Eric Battenberg, RJ Skerry-Ryan, Soroosh Mariooryad, Daisy Stanton, David Kao, Matt Shannon, and Tom Bagby. 2020. Location-relative attention mechanisms for robust long-form speech synthesis. In *Proc. ICASSP*. 6194–6198.

[5] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. *arXiv preprint arXiv:1506.03099* (2015).

[6] Kirsten Bergmann and Manuela Macedonia. 2013. A virtual agent as vocabulary trainer: iconic gestures help to improve learners' memory performance. In *International workshop on intelligent virtual agents*. Springer, 139–148.

[7] David A. Braude. 2016. *Head motion synthesis: Evaluation and a template motion approach*. Ph.D. Dissertation. The University of Edinburgh, UK. http://hdl.handle.net/1842/20418

[8] 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 *Proc. SIGGRAPH*. 413–420.

[9] Justine Cassell, Hannes Högni Vilhjálmsson, and Timothy Bickmore. 2001. BEAT: The behavior expression animation toolkit. In *Proc. SIGGRAPH*. 477–486.

[10] Chung-Cheng Chiu, Louis-Philippe Morency, and Stacy Marsella. 2015. Predicting co-verbal gestures: A deep and temporal modeling approach. In *Proc. IVA*. 152–166.

[11] Ylva. Ferstl and Rachel. McDonnell. 2018. Investigating the use of recurrent motion modelling for speech gesture generation. In *Proc. IVA*. 93–98. https://trinityspeechgesture.scss.tcd.ie

---

[6]See https://swatsw.github.io/isg_icmi21/ or supplemental material for video examples.

[12] Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2019. Multi-objective adversarial gesture generation. In *Proc. MIG*. 3:1–3:10.

[13] Ylva Ferstl, Michael Neff, and Rachel McDonnell. 2020. Adversarial gesture generation with realistic gesture phasing. *Computers & Graphics* 89 (2020), 117–130.

[14] Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. 2019. Learning Individual Styles of Conversational Gesture. In *Proc. CVPR*. 3497–3506.

[15] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks. In *Proc. NIPS*. 2672–2680.

[16] F. Sebastian Grassia. 1998. Practical parameterization of rotations using the exponential map. *Journal of Graphics Tools* 3, 3 (1998), 29–48.

[17] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Lingjie Liu, Hans-Peter Seidel, Gerard Pons-Moll, Mohamed Elgharib, and Christian Theobalt. 2021. Learning Speech-driven 3D Conversational Gestures from Video. *arXiv preprint arXiv:2102.06837* (2021).

[18] Dai Hasegawa, Naoshi Kaneko, Shinichi Shirakawa, Hiroshi Sakuta, and Kazuhiko Sumi. 2018. Evaluation of speech-to-gesture generation using bi-directional LSTM network. In *Proc. IVA*. 79–86.

[19] Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. 2020. MoGlow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics* 39, 4 (2020), 236:1–236:14.

[20] Carlos T. Ishi, Daichi Machiyashiki, Ryusuke Mikata, and Hiroshi Ishiguro. 2018. A speech-driven hand gesture generation method and evaluation in android robots. *IEEE Robotics and Automation Letters* 3, 4 (2018), 3757–3764.

[21] ITU-R BS.1534-3. 2015. *Method for the Subjective Assessment of Intermediate Quality Level of Audio Systems*. Standard. ITU. https://www.itu.int/rec/R-REC-BS.1534-3-201510-I

[22] ITU-T P.800. 1996. *Methods for Subjective Determination of Transmission Quality*. Standard. ITU. https://www.itu.int/rec/T-REC-P.800-199608-I

[23] Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, Taras Kucherenko, and Gustav Eje Henter. 2021. HEMVIP: Human evaluation of multiple videos in parallel. In *Proc. ICMI*.

[24] Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon. 2020. Glow-TTS: A Generative Flow for Text-to-Speech via Monotonic Alignment Search. In *Proc. NeurIPS*. 8067–8077.

[25] Sungwon Kim, Sang-Gil Lee, Jongyoon Song, Jaehyeon Kim, and Sungroh Yoon. 2019. FloWaveNet: A generative flow for raw audio. In *Proc. ICML*. 3370–3378.

[26] Durk P. Kingma and Prafulla Dhariwal. 2018. Glow: Generative flow with invertible 1x1 convolutions. In *Proc. NeurIPS*. 10236–10245.

[27] Michael Kipp. 2005. *Gesture generation by imitation: From human behavior to computer character animation*. Universal-Publishers.

[28] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. In *Proc. NeurIPS*. 17022–17033.

[29] Stefan Kopp and Ipke Wachsmuth. 2004. Synthesizing multimodal utterances for conversational agents. *Computer Animation and Virtual Worlds* 15, 1 (2004), 39–52.

[30] Vladislav Korzun, Ilya Dimov, and Andrey Zharkov. 2020. The FineMotion entry to the GENEA Challenge 2020. In *Proc. GENEA Workshop*. https://doi.org/10.5281/zenodo.4088608

[31] Taras Kucherenko, Dai Hasegawa, Naoshi Kaneko, Gustav Eje Henter, and Hedvig Kjellström. 2021. Moving fast and slow: Analysis of representations and post-processing in speech-driven automatic gesture generation. *International Journal of Human–Computer Interaction* (2021).

[32] Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexanderson, Iolanda Leite, and Hedvig Kjellström. 2020. Gesticulator: A framework for semantically-aware speech-driven gesture generation. In *Proc. ICMI*. 242–250.

[33] Taras Kucherenko, Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, and Gustav Eje Henter. 2021. A large, crowdsourced evaluation of gesture generation systems on common data: The GENEA Challenge 2020. In *Proc. IUI*. 11–21.

[34] Taras Kucherenko, Rajmund Nagy, Michael Neff, Hedvig Kjellström, and Gustav Eje Henter. 2021. Multimodal analysis of the predictability of hand-gesture properties. *arXiv preprint arXiv:2108.05762* (2021).

[35] JinHong Lu, TianHang Liu, ShuZhuang Xu, and Hiroshi Shimodaira. 2021. Double-DCCCAE: Estimation of Body Gestures From Speech Waveform. In *Proc. ICASSP*. 900–904.

[36] Pengcheng Luo, Victor Ng-Thow-Hing, and Michael Neff. 2013. An examination of whether people prefer agents whose gestures mimic their own. In *Proc. IVA*. Springer, 229–238.

[37] Stacy Marsella, Yuyu Xu, Margaux Lhommet, Andrew Feng, Stefan Scherer, and Ari Shapiro. 2013. Virtual character performance from speech. In *Proc. SCA*. 25–35.

[38] Chenfeng Miao, Shuang Liang, Minchuan Chen, Jun Ma, Shaojun Wang, and Jing Xiao. 2020. Flow-TTS: A Non-Autoregressive Network for Text to Speech Based on Flow. In *Proc. ICASSP*. 7209–7213.

[39] Michael Neff, Michael Kipp, Irene Albrecht, and Hans-Peter Seidel. 2008. Gesture modeling and animation based on a probabilistic re-creation of speaker style. *ACM Transactions on Graphics* 27, 1 (2008), 1–24.

[40] Victor Ng-Thow-Hing, Pengcheng Luo, and Sandra Okita. 2010. Synchronized gesture and speech production for humanoid robots. In *Proc. IROS*. 4617–4624.

[41] Wei Ping, Kainan Peng, Kexin Zhao, and Zhao Song. 2020. WaveFlow: A compact flow-based model for raw audio. In *Proc. ICML*. 7706–7716.

[42] Ryan Prenger, Rafael Valle, and Bryan Catanzaro. 2019. WaveGlow: A flow-based generative network for speech synthesis. In *Proc. ICASSP*. 3617–3621.

[43] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog* (2019). https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[44] Manuel Sam Ribeiro, Junichi Yamagishi, and Robert A. J. Clark. 2015. A perceptual investigation of wavelet-based decomposition of $f0$ for text-to-speech synthesis. In *Proc. Interspeech*. 1586–1590.

[45] Jonathan Shen, Ye Jia, Mike Chrzanowski, Yu Zhang, Isaac Elias, Heiga Zen, and Yonghui Wu. 2020. Non-attentive Tacotron: Robust and controllable neural TTS synthesis including unsupervised duration modeling. *arXiv preprint arXiv:2010.04301* (2020).

[46] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, et al. 2018. Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions. In *Proc. ICASSP*. 4779–4783.

[47] Éva Székely, Jens Edlund, and Joakim Gustafson. 2020. Augmented Prompt Selection for Evaluation of Spontaneous Speech Synthesis. In *Proc. LREC*. 6368–6374.

[48] Éva Székely, Gustav Eje Henter, Jonas Beskow, and Joakim Gustafson. 2019. Spontaneous conversational speech synthesis from found data. In *Proc. Interspeech*. 4435–4439.

[49] Éva Székely, Gustav Eje Henter, Jonas Beskow, and Joakim Gustafson. 2020. Breathing and Speech Planning in Spontaneous Speech Synthesis. In *Proc. ICASSP*. 7649–7653.

[50] Éva. Székely, Gustav Eje Henter, Jonas Beskow, and Joakim Gustafson. 2019. How to train your fillers: uh and um in spontaneous speech synthesis. In *Proc. SSW*. 245–250.

[51] Éva Székely, Gustav Eje Henter, and Joakim Gustafson. 2019. Casting to corpus: Segmenting and selecting spontaneous dialogue for TTS with a CNN-LSTM speaker-dependent breath detector. In *Proc. ICASSP*. 6925–6929.

[52] Rafael Valle, Kevin J. Shih, Ryan Prenger, and Bryan Catanzaro. 2021. Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis. In *Proc. ICLR*.

[53] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, et al. 2016. WaveNet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499* (2016).

[54] Petra Wagner, Zofia Malisz, and Stefan Kopp. 2014. Gesture and speech in interaction: An overview. *Speech Communication* 57 (2014), 209–232.

[55] Wenfu Wang, Shuang Xu, and Bo Xu. 2016. First Step Towards End-to-End Parametric TTS Synthesis: Generating Spectral Parameters with Neural Attention. In *Proc. Interspeech*. 2243–2247.

[56] Yuxuan Wang, R. J. Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, et al. 2017. Tacotron: Towards End-to-End Speech Synthesis. In *Proc. Interspeech*. 4006–4010.

[57] Oliver Watts, Gustav Eje Henter, Jason Fong, and Cassia Valentini-Botinhao. 2019. Where do the improvements come from in sequence-to-sequence neural TTS?. In *Proc. SSW*. 217–222.

[58] Yanxiang Wu, Sabarish V Babu, Rowan Armstrong, Jeffrey W Bertrand, Jun Luo, Tania Roy, Shaundra B Daily, Lauren Cairco Dukes, Larry F Hodges, and Tracy Fasolino. 2014. Effects of virtual human animation on emotion contagion in simulated inter-personal experiences. *IEEE Transactions on Visualization and Computer Graphics* 20, 4 (2014), 626–635.

[59] Jin Xu, Xu Tan, Yi Ren, Tao Qin, Jian Li, Sheng Zhao, and Tie-Yan Liu. 2020. LRSpeech: Extremely low-resource speech synthesis and recognition. In *Proc. KDD*. 2802–2812.

[60] Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2020. Speech gesture generation from the trimodal context of text, audio, and speaker identity. *ACM Transactions on Graphics* 39, 6, Article 222 (2020), 16 pages.

[61] 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 *Proc. ICRA*. 4303–4309.

[62] Chengzhu Yu, Heng Lu, Na Hu, Meng Yu, Chao Weng, Kun Xu, Peng Liu, Deyi Tuo, Shiyin Kang, Guangzhi Lei, et al. 2019. DurIAN: Duration informed attention network for multimodal synthesis. *arXiv preprint arXiv:1909.01700* (2019).