# Audio-Driven Stylized Gesture Generation with Flow-Based Model

Sheng Ye[1], Yu-Hui Wen[1], Yanan Sun[1], Ying He[2], Ziyang Zhang[3], Yaoyuan Wang[3], Weihua He[4], and Yong-Jin Liu[1]

[1] CS Dept, BNRist, Tsinghua University
[2] School of Computer Science and Engineering, Nanyang Technological University
[3] Advanced Computing and Storage Lab, Huawei Technologies Co Ltd.
[4] Department of Precision Instrument, Tsinghua University.
{ye-c18,wenyh1616,sunyn20,hwh20,liuyongjin}@tsinghua.edu.cn,
yhe@ntu.edu.sg, {zhangziyang11,wangyaoyuan1}@huawei.com

**Abstract.** Generating stylized audio-driven gestures for robots and virtual avatars has attracted increasing considerations recently. Existing methods require style labels (e.g. speaker identities), or complex preprocessing of data to obtain the style control parameters. In this paper, we propose a new end-to-end flow-based model, which can generate audio-driven gestures of arbitrary styles with neither preprocessing nor style labels. To achieve this goal, we introduce a global encoder and a gesture perceptual loss into the classic generative flow model to capture both global and local information. We conduct extensive experiments on two benchmark datasets: the TED Dataset and the Trinity Dataset. Both quantitative and qualitative evaluations show that the proposed model outperforms state-of-the-art models.

**Keywords:** Stylized Gesture, Flow-based Model, Global Encoder

## 1 Introduction

When people speak, they often make arm and hand movements that accompany what they say. These movements, called co-speech gestures, are important in human communication, as they contain rich non-verbal information [8, 39]. Existing studies have shown that co-speech gestures can help listeners concentrate and better understand the meaning conveyed in oral messages [7, 39]. Therefore, when developing virtual avatars or interactive robots, it is highly desired to generate natural co-speech gestures accompanying their messages, which can improve communication and enhance vividness and realism.

Yet, the problem of audio-driven co-speech gesture synthesis is challenging and intrinsically ill-posed due to the one-to-many relationship between audios and gestures, i.e., the same audio input may correspond to multiple reasonable gestures. Most of the early approaches are rule-based [9, 25, 26, 40], which require complicated mapping rules from audio to motion. These approaches not

Y.H Wen and Y.J Liu are the corresponding authors.

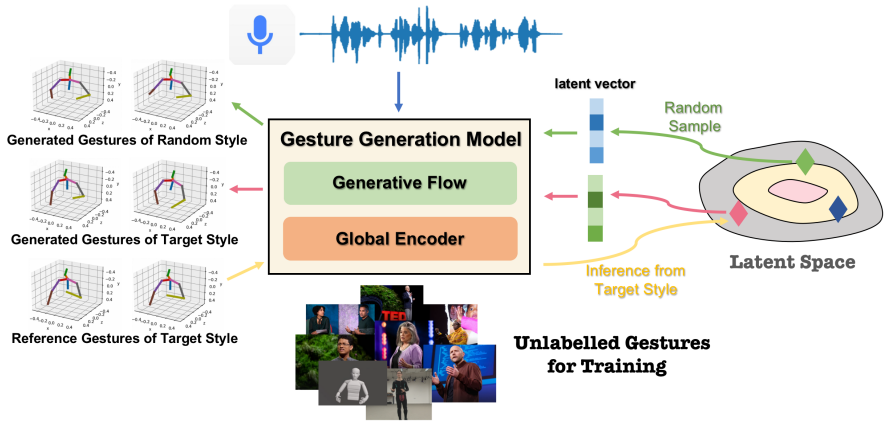


**Fig. 1.** Overview of our proposed stylized audio-driven gesture generation method. We jointly train a generative flow model with a global encoder using unlabelled gesture data. In the synthesis stage, we can manipulate the gesture style given an extra gesture sequence as the target style.

only need a great deal of efforts in designing rules, but also are often too fragile to work for complicated application scenarios. In recent years, data-driven approaches [16, 27, 43] have demonstrated their potential in gesture generation. These methods utilize CNN or RNN based models, which are trained in an end-to-end manner. Since conventional deterministic networks that they use [16, 27, 43] tend to learn one-to-one mapping functions, the generated results often become the average of all potential target gestures and thereby lack diversity.

Furthermore, different people have different styles of co-speech gestures. Previous studies attempt to control the styles of generated gestures. Yoon et al. [42] and Ahuja et al. [2] directly use the speaker's identity as style label and embed it into a latent style space during training. However, it is often difficult to obtain such labelled data in the real-world application scenarios. Alexanderson et al. [5] provide more fine-grained style control, such as movement speed and hand position, but the price is a tedious preprocessing of motion data for obtaining control parameters.

Due to the multi-modal nature of co-speech gestures, we prefer stochastic over deterministic model. Flow-based generative models [10, 18, 24] can generate different plausible results for a single input by randomly sampling the latent distribution, which is much desired for the gesture synthesis task. Therefore in this paper, building upon MoGlow [18], we propose a probabilistic and autoregressive model using normalizing flows to generate audio-driven gestures. Due to widespread availability of unlabelled, real-world gesture data, we propose to manipulate the style of synthesized gestures without the need for style labels during training. Specifically, given an extra target gesture sequence as input, we transfer the style of the target gesture to our generated gesture, which is

similar to the task of style transfer. By assigning different target styles, we can generate co-speech gestures of arbitrary styles. Fig. 1 provides an overview of our proposed method.

Existing generative flow models [10, 18, 24] mainly focus on capturing dependency within local features. In our study, we also draw attention from AutoEncoder which can effectively capture the global feature. In particular, our proposed method jointly train a generative flow model with a global encoder in an AutoEncoder manner to aggregate both the local information and global information. Moreover, we find that only using ordinary $L_1$ or $L_2$ loss cannot effectively measure the difference between the real motion and the generated motion well. Inspired by the success of perceptual losses [22] used in image style transfer tasks, we design a new gesture perceptual loss to help the joint training process and improve the generation quality. We observe that our model can synthesize a large variety of natural and human-like gestures that match the audio input well. We evaluate our proposed model on two benchmark datasets: TED Dataset [43] and Trinity Dataset [12, 29]. Results show that our approach surpasses other state-of-the-art methods in both datasets.

## 2 Related Work

### 2.1 Audio-Driven Gesture Synthesis

Early methods for generating co-speech gestures are typically rule-based [9,25,26] before flourishing development of deep learning methods. Wagner et al. [40] provide a detailed survey of these rule-based methods, which require a great deal of human efforts to design mapping rules. To alleviate human efforts, some data-driven methods are proposed to learn the mapping between prosody features and motions. Specifically, Hidden Markov Models (HMMs) have been used to generate prosody-driven motion sequences [30, 31].

In recent years, VAEs [23] and GANs [15] achieve a great success on image generation problems. These neural network models are also used to predict gesture sequences. Ginosar et al. [14] propose an encoder-decoder structure and train the network with both regression and adversarial losses. Li et al. [32] introduce a conditional VAE model to solve this co-speech gesture generation task.

Classic deep learning models (e.g. CNN or RNN) are deterministic and often suffer from the mean problem when applying to regression problems such as gesture synthesis. Although adversarial learning can reduce this problem to some extent, GAN has its limitations, including intractable log-likelihood and unstable training processes. Henter et al. [18] propose a probabilistic network to model the conditional probability distribution of gesture data, which can not only describe the one-to-many mapping elegantly but also increase the diversity of generated results. However, each flow step of their model only supports linear operations, restricting its expressiveness. Qian et al. [37] complement the audio input with a learnable vector, which reduces ambiguity and turns the one-to-many mapping between the input audio and generated gesture into a one-to-one mapping.

### 2.2 Style Transfer of Motion

Image style transfer, which extracts and transfers the artistic style of one image to another, has been well studied. Gatys et al. [13] first introduce a neural style transfer algorithm. Johnson et al. [22] train a feed-forward network to solve the optimization problem in real time. Huang et al. [21] propose an AdaIN layer to transfer arbitrary styles.

These image-oriented methods are further extended to motion style transfer. Holden et al. [20] introduce a framework that enables motion edition and style transfer. Du et al. [11] propose to use a conditional VAE to learn motion styles, which improve the efficiency of previous approaches. Aberman et al. [1] train their network with unpaired motion data using the AdaIN mechanism and can disentangle motion content and style automatically. Wen et al. [41] propose an unsupervised motion style transfer method using a generative flow model.

Similarly, in gesture synthesis, it is also desired to edit and/or transfer the style of the generated gestures. Though there are previous works on synthesizing such stylized gestures [14,30,35], these methods are only able to learn individual styles. To overcome the challenges, Yoon et al. [42] use speaker identity as an additional input and project it into a style embedding space. By sampling through this space, they can manipulate the gesture styles. Bhattacharya et al. [6] further improve their work [42] by adding an Affective Encoder to learn affective features from the seed poses. Ahuja et al. [2] introduce a supervised learning method that can perform gesture style preservation and style transfer tasks. Another study [5] provides high-level controls over gesturing styles, such as movement speed and spatial extent. But the motion data needs to be preprocessed first to extract control parameters. In contrast, our model is trained on unlabelled gesture data, and can transfer arbitrary styles without any preprocessing procedure, thereby is more suitable for practical applications.

## 3 Approach

### 3.1 Preliminaries on Generative Flow Model

Generative flows, which belongs to generative models, provide advantages such as tractable log-likelihood and efficient inference. Given a set of data X = $\{x_1, x_2, \ldots, x_n\}$, that subject to an unknown distribution, generative flows aim to model the probability distribution $p_\theta(X)$ of X by minimizing the negative log-likelihood:

$$NLL(X) = \frac{1}{n}\sum_{i=1}^{n} -\ln p_\theta(x_i) \tag{1}$$

Unlike VAEs or GANs, generative flows are capable of optimizing this negative log-likelihood exactly. The main idea is to find an invertible and differentiable function $g$ that transforms a simple, fixed distribution $Z$ to a new,

complicated distribution $X$. As the function $g$ is invertible, generative flows can perform efficient sampling as well as efficient inference: $x = g(z), z = g^{-1}(x)$, where $x$ is a data sample and $z$ is a latent variable corresponding to $x$.

The mapping function $g$ can be extremely complex. To increase expressiveness, flow models compose plenty of simple nonlinear transformations $\{g_i\}_{i=1}^{L}$ together to construct the final function: $g = g_1 \circ g_2 \circ ... \circ g_L$. The transformation between two distributions can be described as follows

$$z = h_L \xrightarrow{g_L} h_{L-1} \xrightarrow{g_{L-1}} ... \xrightarrow{g_2} h_1 \xrightarrow{g_1} h_0 = x \tag{2}$$

$$z = g^{-1}(x) = g_L^{-1}(g_{L-1}^{-1}(...g_1^{-1}(x))) \tag{3}$$

$$x = g(z) = g_1(g_2(...g_L(z))). \tag{4}$$

The stacked sequence of inverse transformations $\{g_i^{-1}\}_{i=1}^{L}$ is called normalizing flow. By applying the change-of-variables formula, we can derive the exact log-likelihood of one data sample $x$ as:

$$\ln p_\theta(x) = \ln p_\phi(z) + \sum_{i=1}^{L} \ln \left| \det \left( \frac{\partial h_i}{\partial h_{i-1}} \right) \right|, \tag{5}$$

where $p_\phi(z)$ is the tractable, fixed probability distribution (typically Gaussian distribution or Student's $t$ distribution) and $\det \left( \frac{\partial h_i}{\partial h_{i-1}} \right)$ is the determinant of the Jacobian matrix of $g_i^{-1}$. Calculating the determinant of a dense matrix is computationally expensive. Thus, in usual cases, flow transformations are carefully designed to make the Jacobian matrix diagonal or triangular.

### 3.2 Generative Flow with Global Encoder

Our proposed model can be regarded as an autoregressive sequence-to-sequence model. When generating the $t$-th frame of a gesture sequence $x$, our model is conditioned on $\tau$ frames of previous poses $x_{t-\tau:t-1}$ and $\tau + 1$ frames of audio control signals $a_{t-\tau:t}$. Specifically, all the condition information are fed into every affine coupling layer in the network. Thus, the gesture synthesis procedure can be developed as:

$$p(x|a) = p(x_{1:\tau}|a_{1:\tau}) \prod_{t=\tau+1}^{T} p(x_t|x_{t-\tau:t-1}, a_{t-\tau:t}) \tag{6}$$

Previous studies [19, 33, 34] find that flow-based models have the problem of local dependency, meaning that each flow transformation mainly focuses on capturing dependency within local features. To address this issue, we design an additional global encoder, which is shared by every flow transformation, to provide the global information required by the flow model. The detailed structure of our global encoder can be found in section 3.4. At each time step, this encoder

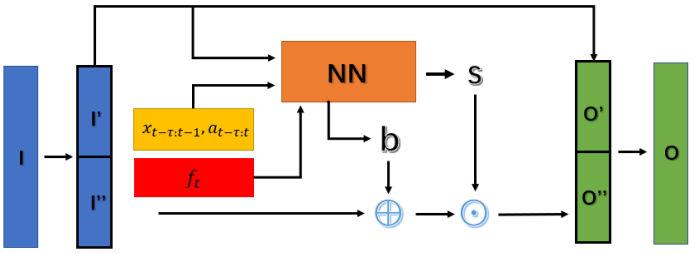


**Fig. 2.** Architecture of the affine coupling layer. We first split the input feature $I$ into two parts, then keep the first part $I^{'}$ unchanged and transform the other $I^{''}$ into $O^{''}$. Finally, we concatenate $O^{''}$ with $O^{'}$ to get the output. $\oplus$ denotes the element-wise sum and $\odot$ is the Hadamard product.

embeds the previous pose histories and audio control signals separately and then integrates them to obtain the final global feature $f_t$. The global feature is then concatenated with original autoregressive poses as well as audio signals and fed into neural networks to produce the bias and scaling parameters $[b_t,\ s_t]$ used in affine coupling layers (shown in Fig. 2). Mathematically, at time step $t$, we denote the input feature and output feature of the affine coupling layer as $I_t$ and $O_t$. The input is first split into two equal parts $[I_t^{'},\ I_t^{''}]$. We keep half of the input unchanged, then shift and scale the other half of the input based on parameters extracted by the neural network of the affine coupling layer. The coupling operation can be defined as:

$$[O_t^{'},\ O_t^{''}] = [I_t^{'},\ (I_t^{''} + b_t) \odot s_t] \tag{7}$$

$$[b_t,\ s_t] = NN(I_t^{'};\ x_{t-\tau:t-1};\ a_{t-\tau:t};\ f_t) \tag{8}$$

where $\odot$ is the Hadamard product. Note that the inverse operation can also be easily obtained

$$[I_t^{'},\ I_t^{''}] = \left[O_t^{'},\ s_t^{-1} O_t^{''} - b_t\right]. \tag{9}$$

As Fig. 3 shows, we propose to jointly train the global encoder with the generative flow in an end-to-end manner. Specifically, we use the invertible flow model as our decoder and train our model like an AutoEncoder. The complete training process can be described as follows. Firstly, we transform the training gesture data into the latent vector conditioned on control signals and global features through the forward normalizing flow. In the meantime, we minimize the NLL loss derived in section 3.1. Then, we sample from the latent, fixed distribution randomly to get the new latent vector and transform it into the synthesized gesture pose conditioned on the same global features and controls through the backward normalizing flow. We simultaneously minimize the reconstruction loss $L_{recon}$, which measures the $L_1$ distance between the real motion and the synthesized motion. Through the pass of the forward flow, we train the flow model to

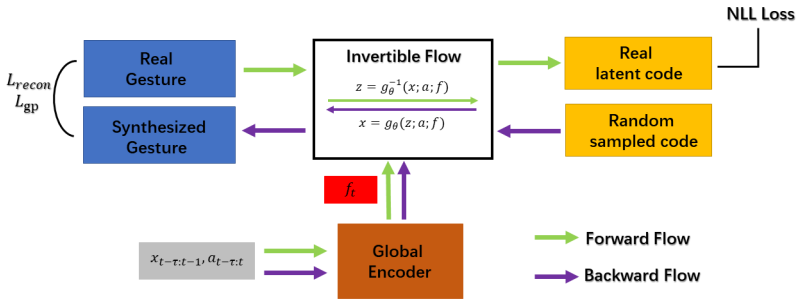


**Fig. 3.** Overview of our proposed training framework. Through the pass of the forward flow and backward flow, we can jointly train the global encoder with the generative flow to complement each other in an end-to-end manner.

take full advantage of global features. Through the pass of the backward flow, we train the global encoder to capture meaningful features. See the supplementary material for the detailed structure of the invertible flow.

### 3.3 Gesture Peceptual Loss

Our reconstruction loss $L_{recon}$ computes the difference between the generated gesture and the ground-truth gesture. However, the low-level $L_1$ loss is insufficient to measure the difference well. Inspired by the image perceptual losses [22], we further propose a new gesture perceptual loss to train our model. To the best of our knowledge, there is no pre-trained inception network for gesture data (like VGG-Nets [38] for images). Therefore, we train our own feature extraction net using the training set. Our feature extractor has a convolutional encoder-decoder structure. Denote by $x$ the real gesture motion, $\hat{x}$ the synthesized gesture motion, and $\omega$ the 4-th layer of our feature extractor encoder. As $\omega$ is a convolutional layer, $\omega(x)$ and $\omega(\hat{x})$ are feature maps of dimension $C \times H \times W$. Thus, we can describe our proposed loss as:

$$L_{gp}(\hat{x}, x) = \frac{1}{C \times H \times W} \|\omega(\hat{x}) - \omega(x)\|_1. \tag{10}$$

Note that this loss encourages synthesized motion and real motion to have similar feature representations. Our ablation study further demonstrates that this loss facilitates the joint training process of our model and improves the quality of generated results. Finally, we train our model with the combined loss:

$$L = NLL + \lambda_1 L_{recon} + \lambda_2 L_{gp}, \tag{11}$$

where $\lambda_1$ and $\lambda_2$ are the pre-defined weights. In our experiments, we empirically set $\lambda_1 = 1.0$ and $\lambda_2 = 0.1$.

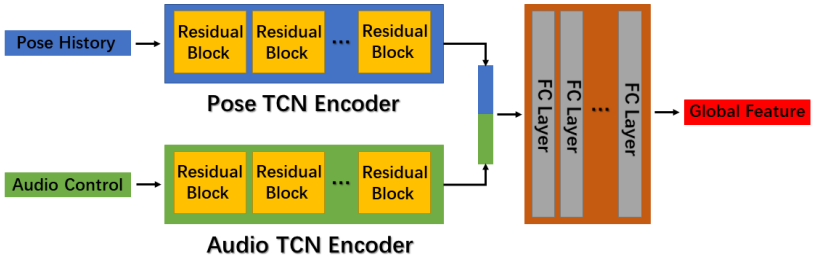


**Fig. 4.** Structure of proposed global encoder. Autoregressive pose histories and audio control signals are encoded as low-dimensional vectors by two TCN encoders, respectively. The two vectors are then concatenated together and passed through a series of fully connected layers to get the final global feature.

### 3.4 Network Structure

Our neural network builds upon the MoGlow [5] network. Specifically, we keep the structure of Actnorm layers and invertible $1 \times 1$ convolutions. As for affine coupling layers, we replace LSTMs with GRUs. We observe that in this task, GRU can generate results comparable to LSTM, but is more concise and computationally efficient, and has a faster convergence rate due to the fewer parameters. Besides, the previous MoGlow model with LSTM has a limitation in that the generated results rely more heavily on autoregressive histories than audio control signals. A possible reason for this is the overly complex gating mechanism of LSTM, which makes the network retain too much redundant information from previous states. Our proposed model with GRU can alleviate this problem to some extent. We also adapt the zero initialization technique proposed by Kingma et al. [24], making each affine coupling layer performs the identity function at the initial state.

Our global encoder (shown in Fig. 4) consists of a series of fully connected layers and two TCN (Temporal Convolution Network) encoders stacked by many residual building blocks. Each building block is initialized by Kaiming Normal [17] and includes four basic operations: a 1D convolution, a batch normalization, a residual operation, and a ReLU activation. Empirically, we assume that this architecture is designed to extract the global features of gestures and audio sequences and the correlations between them. Furthermore, the global encoder can be trained efficiently as it is based on temporal convolutions.

## 4 Experiments

We implement our network using PyTorch. Our model is optimized using Adam on an NVIDIA A100 GPU. We choose the Student's $t$ distribution as the fixed latent distribution as it provides a more robust training process [4]. We conduct our experiments on two benchmark datasets (described in detail in Section 4.1), namely, TED Dataset and Trinity Dataset. For the TED Dataset, our model is

trained for 80k iterations with a batch size of 200. For the Trinity Dataset, our model is trained for 30k iterations with a batch size of 100.

### 4.1 Datasets

**TED Dataset**
TED Dataset, which is first proposed by Yoon et al. [43], is a large, English-language dataset for gesture synthesis problems. It contains upper body pose sequences, audio waveforms, transcribed speech texts, and speaker identities. Pose data is extracted from 1,714 videos of TED lectures and converted to 3D data by using a pose estimator [36]. We resample the pose data at 15 fps and slice each training sample into 42 frames. We use the initial 12 frames as autoregressive pose histories and, following Yoon et al. [42], train our model to generate the remaining 30 frames. The dataset is partitioned into a training set (200,038 samples), a test set (26,245 samples), and a validation set (26,903 samples). We use the training set to train our model, the test set for quantitative and qualitative evaluation, and the validation set for tuning our network.

**Trinity Dataset**
Trinity Dataset (GENEA Challenge 2020 [29]) proposed by Ferstl et al. [12] is a dataset containing speaking gestures and corresponding audio signals. Unlike TED Dataset, this dataset only consists of a single speaker and is collected using a professional motion capture system, which results in better data quality. Moreover, this dataset contains full-body motion, including the speaker changing the standing posture or taking a few steps back and forth. Following Alexanderson et al. [5], we downsample the motion data at 20 fps and slice each training sample into 120 frames. The total duration of valid data is about 242 minutes and *Recording_008* session is held out for validation and evaluation. We take 5 frames as historic poses and 20 frames as audio lookahead. We also augment the data by mirroring the gesture motion with the speech unchanged.

### 4.2 Evaluation

**Quantitative Evaluation**
We compare our model with several state-of-the-art models on two datasets using different metrics. On the TED Dataset, we compare with six representative methods: attentional Seq2Seq (Seq2Seq) [43], Speech2Gesture (S2G) [14], Joint Embedding Model (JEM) [3], MoGlow [5], Gestures from Trimodal Context (GTC) [42], and Speech2AffectiveGestures (S2AG) [6]. On the Trinity Dataset, we compare with two methods: Gesticulator [28] and MoGlow [5]. We use Fréchet Gesture Distance (FGD), Percent of Correct Keypoints (PCK), Diversity (Div), BeatAlign Score (BA), and Multi-modality Score (MM) for evaluation. Please refer to our supplementary material for detailed descriptions of these metrics.

For Seq2Seq, S2G, JEM, we directly use some metric values as documented in [42]. For GTC, S2AG, and Gesticulator, we utilize the pre-trained models

**Table 1.** Comparison of our method with previous methods on two benchmark datasets. Bold indicates the best.

| Dataset | Method | FGD | PCK | Div | BA($\sigma = 5$) | MM |
|---|---|---|---|---|---|---|
| TED | Seq2Seq [43] | 18.15 | 0.809 | 40.08 | 0.319 | - |
| | S2G [14] | 19.25 | 0.877 | 44.46 | 0.620 | - |
| | JEM [3] | 22.08 | **0.880** | 44.61 | 0.508 | - |
| | MoGlow [5] | 5.15 | 0.842 | 50.63 | 0.617 | 22.56 |
| | GTC [42] | 4.40 | 0.850 | 48.70 | 0.618 | - |
| | S2AG [6] | 7.22 | 0.861 | 47.25 | 0.619 | - |
| | Ours | **3.30** | 0.850 | **52.00** | **0.622** | **22.63** |
| Trinity | Gesticulator [28] | 9.95 | 0.701 | 3.50 | 0.844 | - |
| | MoGlow [5] | 7.61 | 0.726 | 3.81 | 0.852 | **2.12** |
| | Ours | **6.76** | **0.730** | **4.34** | **0.875** | 1.60 |

provided by the authors. We train MoGlow from scratch following the same configuration as in [5].

Table 1 summarizes the comparison results. On the TED Dataset, our method achieves the best performance for almost all the metrics except PCK, indicating that our generated gestures are high quality. Specifically, for FGD metric, we achieve improvements of 25.0% and 54.3% over GTC and S2AG, which both need trimodal contexts. Note that JEM, S2G, and S2AG are better than our model in the PCK metric. A possible reason is that our model generates plausible gestures from a probabilistic model, while the other methods tend to generate averaged motion which lacks diversity. On the Trinity Dataset, our model also surpasses two other state-of-the-art models. Although MoGlow achieves better MM scores than ours, we emphasize that higher MM is only preferred when the generated gestures are realistic and natural, because invalid and jitter motion can also result in high MM scores.

**Qualitative Evaluation**

Fig. 5 shows the qualitative results of our method on the TED Dataset. The synthesized poses are plotted as stick figures. We also demonstrate the full-body generation results of the Trinity Dataset in Fig. 6. For better visualization, we retarget the motion sequences to several 3D characters using Blender. Note that the generated gestures are diverse and human-like, and match the input audio well.

**User Study**

To further evaluate the quality of our results based on human perception, we conducted a user study with 12 participants. Specifically, we asked the participants to watch six groups of videos, where each group contained three gesture sequences synthesized by GTC [42], S2AG [6], and our method, respectively. They were asked to rate each gesture sequence in terms of Realism, Matching

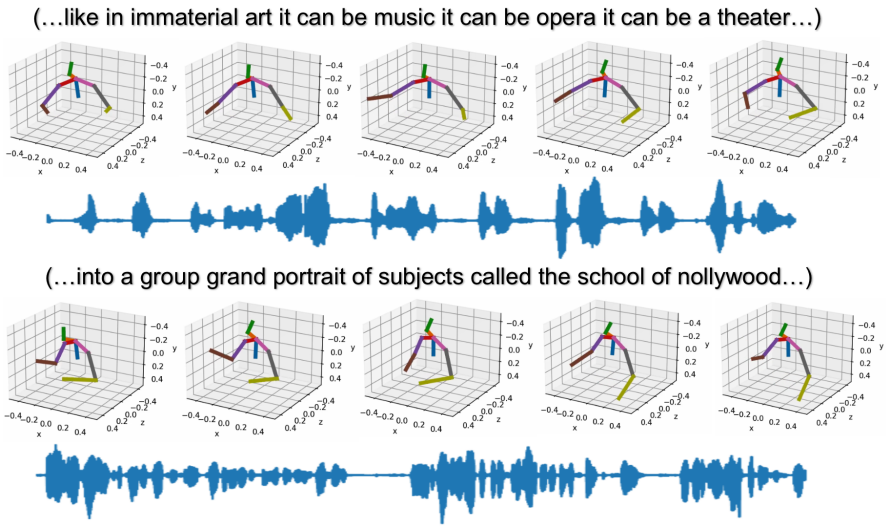


**Fig. 5.** Qualitative results (visualized as 3D stick figures) of our method on the TED Dataset. We also show the corresponding audio waveforms and texts.

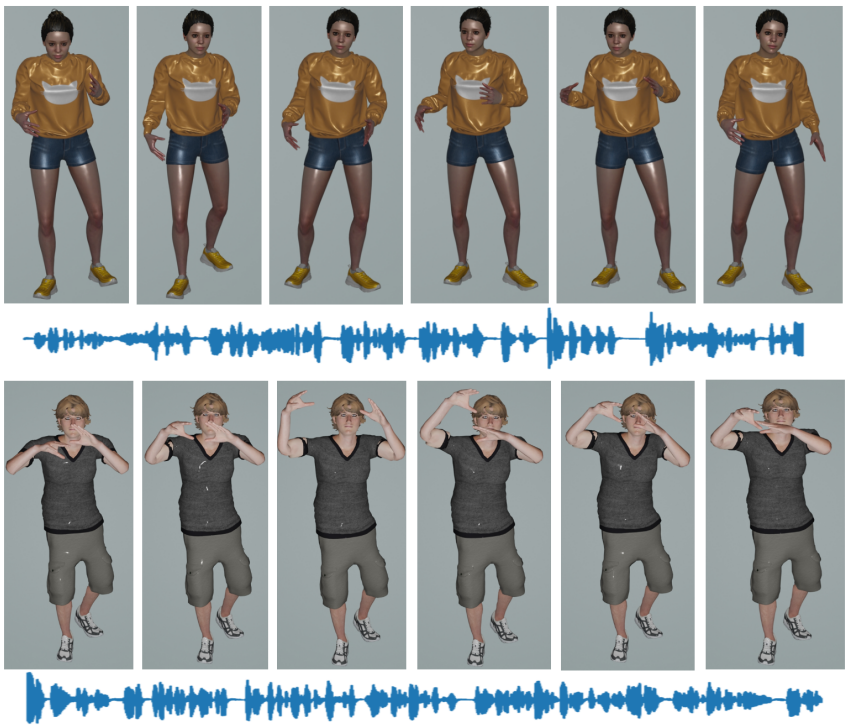

**Fig. 6.** Full-body generated poses and corresponding audio waveform of the Trinity Dataset. We retarget the pose sequences to 3D characters using Blender.

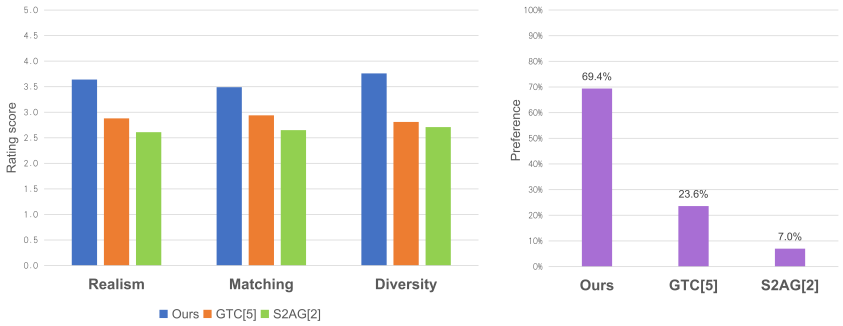


**Fig. 7.** User study results of comparing our method with two state-of-the-art methods on TED Dataset, i.e., GTC [42], and S2AG [6]. The left figure shows human rating scores on three aspects: Realism, Matching Degree, and Diversity. The right figure shows human preference over results generated by different method.

Degree, and Diversity. Moreover, the participants were required to choose one gesture sequence in each group that they liked the most.

Fig. 7 shows the user study results. The left histogram demonstrates that our proposed method outperforms the other two state-of-the-art methods. The right histogram shows that the majority of participants (almost 70%) preferred our synthesized gestures, again confirming that our results are more visually appealing than the others. See also the supplementary material and demo video for more results.

### 4.3 Ablation Study

We conduct an ablation study on TED Dataset to understand each part of our model in detail. Specifically, we remove two components of the proposed model separately: the global encoder and the gesture perceptual loss, and measure FGD, PCK, and Div metrics. Table 2 shows the results of our ablation study.

By removing both global encoder and gesture perceptual loss, our model are degenerate and become MoGlow, except that the LSTMs are replaced with GRUs. We take this as our baseline. Without $L_{gp}$ or global encoder, the FGD and Div scores get worse, implying that the results are less natural and lack diversity. Note that our proposed gesture perceptual loss is used to facilitate the joint training process of the flow model and the global encoder. Therefore, adding $L_{gp}$ without the global encoder makes little sense and results in a slightly worse performance than the baseline (Table 2, third row). As our proposed global encoder and gesture perceptual loss are mainly used to enhance the global features of gestures, thus, the improvement of our model under the FGD metric is more significant than PCK and Div metrics. The ablation study confirms that both the global encoder and the gesture perceptual loss have positive effects on co-speech gesture generation.

**Table 2.** Results of our ablation study. For FGD, lower values are better. For PCK and Div, higher values are better.

| Config | FGD | PCK | Div(mm) |
|---|---|---|---|
| Proposed (no ablation) | 3.30 | 0.850 | 52 |
| Without $L_{gp}$ | 4.62 | 0.845 | 51 |
| Without Global Encoder | 6.26 | 0.848 | 50 |
| Baseline | 6.10 | 0.850 | 49 |

### 4.4 Latent Space Visualization

The GAN-based models usually lack latent representation of the data samples. In contrast, normalizing flows can directly transform a gesture to its corresponding latent code through the pass of the forward flow. By projecting these latent codes onto a 2D space using t-SNE and coloring each sample according to its style label (speaker's identity), we can visualize the distribution of the latent space of gestures. We observe that the models trained on the full TED Dataset tend to extract general features and thereby cannot distinguish different samples well in the latent space. Therefore, we train our model on a subset (with only 15 speakers) of the TED Dataset and visualize the results in Fig. 8 left.

**Table 3.** Comparison of the clustering results of our method and MoGlow on a subset (15 speakers) of the TED Dataset. Higher CHI and SCoeff values indicate better performance.

| Method | CHI | SCoeff |
|---|---|---|
| MoGlow [5] | 908.47 | 0.443 |
| Ours | **1149.34** | **0.517** |

We observe that latent codes are clustered into different groups, and samples from the same speaker tend to be in adjacent spaces. It means that although our model does not need labels and is trained in an unsupervised manner, it can still learn the styles of gestures and can encode gestures with similar styles into nearby locations in the latent space. Moreover, compared to MoGlow, we find that latent codes inferred by our model are clustered more reasonably and the gap between different categories is more significant. Mathematically, we compute the Calinski-Harabaz Index (CHI) and the Silhouette Coefficient (SCoeff) to measure the clustering result. Table 3 shows that our model can learn the distribution of gesture styles better.

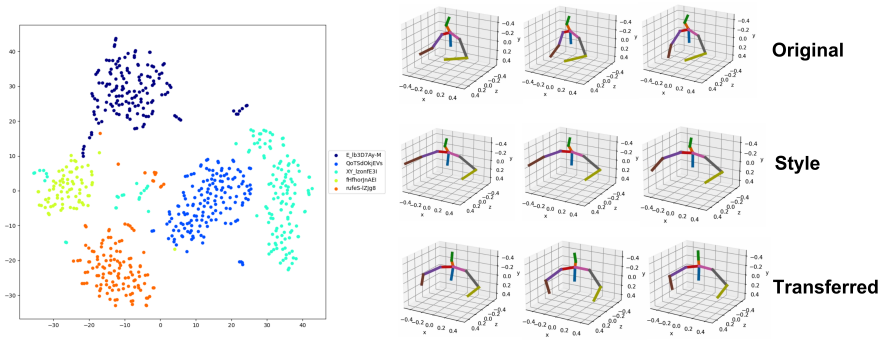


**Fig. 8.** Visualization of the latent space (left) and a typical example of gesture style transfer (right). We project the latent codes to $\mathbb{R}^2$ and color them based on their labels. Gestures are transferred to a drastic and exaggerated style.

### 4.5 Manipulating Gesture Styles

Section 4.4 has demonstrated that the latent codes contain high-level properties of gesture styles. Thus, by manipulating the latent codes during inference time, we can control the style of synthesized gestures. Specifically, when generating the stylized gesture, we replace the randomly sampled latent code with the specific latent code inferred from the target style gesture sequence. Therefore, we can generate audio-driven gestures with a specific style.

Fig. 8 (right) shows such an example. The original poses are gentle and restrained with slight arm movement (first row). However, we want to generate passionate and drastic motions (second row). The transferred results (third row) show exaggerated gestures with widely open arms. Furthermore, the rhythm of the transferred gestures is consistent with the original movements.

## 5 Conclusions

In this paper, we propose an end-to-end flow-based model to synthesize stylized audio-driven gestures in an unsupervised manner. Our model is novel in that it utilizes a global encoder to capture both the local and global features by jointly training the encoder and the generative flow. We also design a gesture perceptual loss to facilitate the joint training procedure and improve the quality of the generated results. Both quantitative and qualitative evaluations on two benchmark datasets show that the proposed approach outperforms state-of-the-art methods. Moreover, the flow-based model allows us to explore the latent space to transfer the styles of target gestures to our generated gestures.

**Acknowledgments:** This work was supported by the Natural Science Foundation of China (No.61725204), Tsinghua University Initiative Scientific Research Program, China Postdoctoral Science Foundation (No.2021M701891).

## References


1. Aberman, K., Weng, Y., Lischinski, D., Cohen-Or, D., Chen, B.: Unpaired motion style transfer from video to animation. ACM Transactions on Graphics (TOG) **39**(4), 64–1 (2020)
2. Ahuja, C., Lee, D.W., Nakano, Y.I., Morency, L.P.: Style transfer for co-speech gesture animation: A multi-speaker conditional-mixture approach. In: European Conference on Computer Vision. pp. 248–265. Springer (2020)
3. Ahuja, C., Morency, L.P.: Language2pose: Natural language grounded pose forecasting. In: 2019 International Conference on 3D Vision (3DV). pp. 719–728. IEEE (2019)
4. Alexanderson, S., Henter, G.E.: Robust model training and generalisation with studentising flows. In: ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models (INNF+ 2020). vol. 2, pp. 25–1 (2020)
5. Alexanderson, S., Henter, G.E., Kucherenko, T., Beskow, J.: Style-controllable speech-driven gesture synthesis using normalising flows. In: Computer Graphics Forum. vol. 39, pp. 487–496. Wiley Online Library (2020)
6. Bhattacharya, U., Childs, E., Rewkowski, N., Manocha, D.: Speech2affectivegestures: Synthesizing co-speech gestures with generative adversarial affective expression learning. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 2027–2036 (2021)
7. Bremner, P., Pipe, A.G., Melhuish, C., Fraser, M., Subramanian, S.: The effects of robot-performed co-verbal gesture on listener behaviour. In: 2011 11th IEEE-RAS International Conference on Humanoid Robots. pp. 458–465. IEEE (2011)
8. Cassell, J., McNeill, D., McCullough, K.E.: Speech-gesture mismatches: Evidence for one underlying representation of linguistic and nonlinguistic information. Pragmatics & cognition **7**(1), 1–34 (1999)
9. Cassell, J., Pelachaud, C., Badler, N., Steedman, M., Achorn, B., Becket, T., Douville, B., Prevost, S., Stone, M.: Animated conversation: rule-based generation of facial expression, gesture & spoken intonation for multiple conversational agents. In: Proceedings of the 21st annual conference on Computer graphics and interactive techniques. pp. 413–420 (1994)
10. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real nvp. In: International Conference on Learning Representations (2017)
11. Du, H., Herrmann, E., Sprenger, J., Cheema, N., Hosseini, S., Fischer, K., Slusallek, P.: Stylistic locomotion modeling with conditional variational autoencoder. In: Eurographics (Short Papers). pp. 9–12 (2019)
12. Ferstl, Y., McDonnell, R.: Investigating the use of recurrent motion modelling for speech gesture generation. In: Proceedings of the 18th International Conference on Intelligent Virtual Agents. pp. 93–98 (2018)
13. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2414–2423 (2016)
14. Ginosar, S., Bar, A., Kohavi, G., Chan, C., Owens, A., Malik, J.: Learning individual styles of conversational gesture. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3497–3506 (2019)
15. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014)

16. Hasegawa, D., Kaneko, N., Shirakawa, S., Sakuta, H., Sumi, K.: Evaluation of speech-to-gesture generation using bi-directional lstm network. In: Proceedings of the 18th International Conference on Intelligent Virtual Agents. pp. 79–86 (2018)
17. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
18. Henter, G.E., Alexanderson, S., Beskow, J.: Moglow: Probabilistic and controllable motion synthesis using normalising flows. ACM Transactions on Graphics (TOG) **39**(6), 1–14 (2020)
19. Ho, J., Chen, X., Srinivas, A., Duan, Y., Abbeel, P.: Flow++: Improving flow-based generative models with variational dequantization and architecture design. In: International Conference on Machine Learning. pp. 2722–2730. PMLR (2019)
20. Holden, D., Saito, J., Komura, T.: A deep learning framework for character motion synthesis and editing. ACM Transactions on Graphics (TOG) **35**(4), 1–11 (2016)
21. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE international conference on computer vision. pp. 1501–1510 (2017)
22. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)
23. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: International Conference on Learning Representations (2014)
24. Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. Advances in neural information processing systems **31** (2018)
25. Kipp, M.: Gesture generation by imitation: From human behavior to computer character animation. Universal-Publishers (2005)
26. Kopp, S., Krenn, B., Marsella, S., Marshall, A.N., Pelachaud, C., Pirker, H., Thórisson, K.R., Vilhjálmsson, H.: Towards a common framework for multimodal generation: The behavior markup language. In: International workshop on intelligent virtual agents. pp. 205–217. Springer (2006)
27. Kucherenko, T., Hasegawa, D., Henter, G.E., Kaneko, N., Kjellström, H.: Analyzing input and output representations for speech-driven gesture generation. In: Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents. pp. 97–104 (2019)
28. Kucherenko, T., Jonell, P., van Waveren, S., Henter, G.E., Alexandersson, S., Leite, I., Kjellström, H.: Gesticulator: A framework for semantically-aware speech-driven gesture generation. In: Proceedings of the 2020 International Conference on Multimodal Interaction. pp. 242–250 (2020)
29. Kucherenko, T., Jonell, P., Yoon, Y., Wolfert, P., Henter, G.E.: A large, crowdsourced evaluation of gesture generation systems on common data: The genea challenge 2020. In: 26th International Conference on Intelligent User Interfaces. pp. 11–21 (2021)
30. Levine, S., Krähenbühl, P., Thrun, S., Koltun, V.: Gesture controllers. In: ACM SIGGRAPH 2010 papers, pp. 1–11. Association for Computing Machinery, New York, NY, United States (2010)
31. Levine, S., Theobalt, C., Koltun, V.: Real-time prosody-driven synthesis of body language. In: ACM SIGGRAPH Asia 2009 papers, pp. 1–10. Association for Computing Machinery, New York, NY, United States (2009)
32. Li, J., Kang, D., Pei, W., Zhe, X., Zhang, Y., He, Z., Bao, L.: Audio2gestures: Generating diverse gestures from speech audio with conditional variational autoen-

coders. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11293–11302 (2021)
33. Ma, X., Kong, X., Zhang, S., Hovy, E.: Macow: Masked convolutional generative flow. Advances in Neural Information Processing Systems **32** (2019)
34. Ma, X., Kong, X., Zhang, S., Hovy, E.H.: Decoupling global and local representations via invertible generative flows. In: International Conference on Learning Representations (2020)
35. Neff, M., Kipp, M., Albrecht, I., Seidel, H.P.: Gesture modeling and animation based on a probabilistic re-creation of speaker style. ACM Transactions on Graphics (TOG) **27**(1), 1–24 (2008)
36. Pavllo, D., Feichtenhofer, C., Grangier, D., Auli, M.: 3d human pose estimation in video with temporal convolutions and semi-supervised training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7753–7762 (2019)
37. Qian, S., Tu, Z., Zhi, Y., Liu, W., Gao, S.: Speech drives templates: Co-speech gesture synthesis with learned templates. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11077–11086 (2021)
38. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (2015)
39. Studdert-Kennedy, M.: Hand and mind: What gestures reveal about thought. Language and Speech **37**(2), 203–209 (1994)
40. Wagner, P., Malisz, Z., Kopp, S.: Gesture and speech in interaction: An overview. Speech Communication **57**, 209–232 (2014)
41. Wen, Y.H., Yang, Z., Fu, H., Gao, L., Sun, Y., Liu, Y.J.: Autoregressive stylized motion synthesis with generative flow. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13612–13621 (2021)
42. Yoon, Y., Cha, B., Lee, J.H., Jang, M., Lee, J., Kim, J., Lee, G.: Speech gesture generation from the trimodal context of text, audio, and speaker identity. ACM Transactions on Graphics (TOG) **39**(6), 1–16 (2020)
43. Yoon, Y., Ko, W.R., Jang, M., Lee, J., Kim, J., Lee, G.: Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 4303–4309. IEEE (2019)