Trinity College Dublin, The University of Dublin

Machine Learning For Plausible Gesture Generation From Speech For Virtual Humans

by

Ylva Ferstl

Under the supervision of Dr. Rachel McDonnell

A thesis submitted to
Trinity College Dublin, The University of Dublin
for the degree of
Doctor of Philosophy

in the
Graphics Vision and Visualisation Group
School of Computer Science and Statistics

August 2021

Declaration of Authorship

I, YLVA FERSTL, declare that this thesis titled, 'Machine Learning For Plausible Gesture Generation From Speech For Virtual Humans' and the work presented in it are my own. I confirm that:

- I declare that this thesis has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work.
- I agree to deposit this thesis in the University's open access institutional repository or allow the Library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.
- I consent to the examiner retaining a copy of the thesis beyond the examining period, should they so wish (EU GDPR May 2018).
- This work was done wholly while in candidature for a research degree at this University.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Ylva Ferstl
February 2021

Trinity College Dublin, The University of Dublin

Abstract

Machine Learning For Plausible Gesture Generation From Speech For Virtual Humans

by Ylva Ferstl

Under the supervision of Dr. Rachel McDonnell

The growing use of virtual humans in an array of applications such as games, human-computer interfaces, and virtual reality demands the design of appealing and engaging characters, while minimizing the cost and time of creation. Nonverbal behavior is an integral part of human communication and important for believable embodied virtual agents. Co-speech gesture represents a key aspect of nonverbal communication and virtual agents are more engaging when exhibiting gesture behavior. Hand-animation of gesture is costly and does not scale to applications where agents may produce new utterances after deployment. Automatized gesture generation is therefore attractive, enabling any new utterance to be animated on the go. A major body of research has been dedicated to methods of automatic gesture generation, but generating expressive and defined gesture motion has commonly relied on explicit formulation of if-then rules or probabilistic modelling of annotated features. Able to work on unlabelled data, machine learning approaches are catching up, however, they often still produce averaged motion failing to capture the speech-gesture relationship adequately. The results from machine-learned models point to the high complexity of the speech-to-motion learning task. In this work, we explore a number of machine learning methods for improving the speech-to-motion learning outcome, including the use of transfer learning from speech and motion models, adversarial training, as well as modelling explicit expressive gesture parameters from speech. We develop a method for automatically segmenting individual gestures from a motion stream, enabling detailed analysis of the speech-gesture relationship. We present two large multimodal datasets of conversational speech and motion, designed specifically for this modelling problem. We finally present and evaluate a novel speech-to-gesture system, merging methods of machine learning and database sampling.

Acknowledgements

Firstly, I would like to thank my supervisor Dr. Rachel McDonnell for taking me on and giving her support throughout the time of this work. She provided much needed encouragement and positivity while helping me develop my academic skills. I appreciate the freedom I was given to take academic and professional opportunities, as well as the freedom in shaping this project.

Secondly, I want to thank Dr. Michael Neff for enabling and co-guiding much of the research contained in this thesis. I am grateful for the significant time and many resources I was given for a rewarding academic stay, as well as his support in gaining industry experience.

Finally, a thank you to the members of my PhD committee for giving their time and making the virtually held viva voce an enjoyable experience: Dr. Stacy Marsella and Dr. Michael Manzke kindly reviewed this thesis and served as examiners, and Dr. Douglas Leith chaired the examination.

This research was funded by Science Foundation Ireland under the ADAPT Centre for Digital Content Technology (Grant 13/RC/2016).

Contents

List of Figures

List of Tables

List of Relevant Publications

1. Ferstl, Ylva, and Rachel McDonnell. "Investigating the use of recurrent motion modelling for speech gesture generation." In *Proceedings of the 18th International Conference on Intelligent Virtual Agents (IVA)*, pp. 93-98. 2018.

2. Ferstl, Ylva, Michael Neff, and Rachel McDonnell. "Multi-objective adversarial gesture generation." In *Motion, Interaction and Games (MIG)*, pp. 1-10. 2019.

3. Ferstl, Ylva, Michael Neff, and Rachel McDonnell. "Adversarial gesture generation with realistic gesture phasing." *Computers & Graphics* 89 (2020): 117-130.

4. Ferstl, Ylva, Michael Neff, and Rachel McDonnell. "Understanding the predictability of gesture parameters from speech and their perceptual importance." In *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents (IVA)*, pp. 1-8. 2020.

5. Ferstl, Ylva, Michael Neff, and Rachel McDonnell. "It's A Match! Gesture Generation Using Expressive Parameter Matching (Extended Abstract)." In *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1495-1497. 2021.

6. Ferstl, Ylva, Michael Neff, and Rachel McDonnell. "ExpressGesture: Expressive Gesture Generation from Speech through Database Matching". In *Computer Animation and Virtual Worlds (CAVW)*. 2021.

Chapter 1

Introduction

Virtual humans are becoming increasingly popular for many applications, such as video games, human-computer interfaces (e.g., virtual museum guides [5]), virtual reality entertainment, and personalized training (e.g., virtual patients for medical training [6]), including training of interpersonal skill, and people may enjoy interacting with them more than even with realistic video-based characters [7]. However, they often still feel stiff and unnatural. Non-verbal behavior plays an important role in making these agents more appealing, and co-speech gestures specifically are a key component for increasing user engagement [8].

Users detect whether virtual agents' gestures are consistent with the produced speech [9] and realistic gestures are essential for adequately mimicking real human interactions, in which non-verbal behaviour plays a major role in conveying information [10, 11]. Co-speech gesture behavior also influences user's perceptions of personality [12, 13] and competence [14] of the virtual agent, emphasizing the important role of gesture in agent design.

Producing realistic gesture behavior for virtual agents is a non-trivial problem. To remove the need for tedious hand-animation, various approaches have been proposed to automatically generate gesture animations from speech, including rule-based systems, statistical models, and machine learning works, each coming with advantages as well as caveats.

Rule-based systems produce defined, exact gesture form, as well as being able to incorporate semantically meaningful gestures through their explicit phrase-to-gesture rules.

Their design does not require actual recording of speech and gesture, rather, hand-crafted animations can be used and explicitly associated with speech markers. The speech input is often required as a text transcript rather than the only audio signal in order to allow for semantic analysis. Designing the rules for a system can be tedious work and hence the expressivity of the system can be limited.

Statistical models on the other hand rely on modelling actual collected conversational data. They estimate conditional probabilities of specific speech features co-occurring with a set of defined motion features within the data. Most such approaches rely on hand-annotation of speech and/or motion features. Statistical models may work with relatively smaller datasets than more automatic machine learning approaches, however, they are also limited in using larger datasets if using hand-annotation.

Machine learning approaches can utilize large and unstructured datasets and produce novel motion not seen in the training data. However, as they aim to capture relationships between speech and motion implicitly through many examples, they are rarely able to produce any semantically meaningful gestures. The produced motion can also lack definition and form, and a large dataset is a requirement.

One major challenge in modelling gesture motion is the large variability of gesture, with gesture choice and expression varying both between speakers as well as within speaker. The same utterance may be accompanied by two completely different gestures even when repeated by the same speaker at different points in time. Rather than speech directly informing the gestures to be produced, the *Growth Point* theory of McNeill [15] argues that speech and gesture are both expressions of the same cognitive process, two channels expressing the same idea. Therefore, speech may give us an indication of the underlying intention that inspired a gesture, but may never fully predict the gesture expression.

1.1 Motivation

The motivation of this work stemmed from the lack of satisfying systems for automatically generating gesture motion from a speech audio. We wanted to harness the power of new machine learning methods of being able to learn from large amounts of unlabelled data; this gives the advantage of easy extensibility and improvement when more speaker data becomes available, ensuring long-term usability of the system. Secondly, we wanted

to address the problem of averaged, unappealing motion often resulting from mean pose convergence in standard regression training of machine learning models. Addressing the non-deterministic and highly variable relationship of speech and gesture, we wanted to avoid modelling a specific 'correct' gesture for an utterance and were instead interested in generating *plausible* gesture behavior, gestures perceived by the observer to match the speech expression.

1.2 Methodology

For modelling relationship of speech and gesture, we made use of a number of machine learning methods, described below.

1.2.1 Recurrent Neural Networks

Recurrent neural networks (RNNs) simulate situational memory that lets prior inputs influence the output of the current input. (Situational memory is hereby opposed to the general "memory" of a network, the connection weights set through the training process of a neural network.) This situational memory is realized by cells maintaining states; cells can add or remove information from this state to keep it up to date for the current context. Long Short Term Memory (LSTM) and Gated Recurrent Units (GRUs) are common variants of recurrent cells, with GRUs being a simpler variant that can be trained faster.

RNNs are often used for a so-called sequence-to-sequence architecture. Here, an RNN layer encodes an input sequence and yields its internal cell state. This encoder can also be a stack of recurrent layers, in which case the last layer yields its internal state. This internal state is the input to the decoder. The decoder equally is an RNN layer (or stack thereof) and it produces an output sequence. For example, an input sequence to the encoder could be a sentence in English, and the output of the decoder the same sentence in French. This example is illustrated in Figure 1.1: a sequence-to-sequence model using LSTM cells translates "How are you" to French.

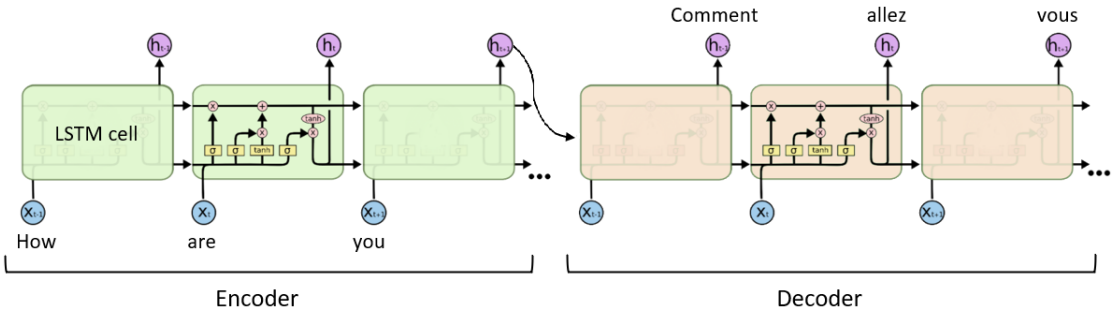


FIGURE 1.1: Example of a sequence-to-sequence model using LSTM cells. The LSTM cell as three gates (marked as σ) to maintain and use the hidden cell state h: An input, forget, and output gate. (Image adapted from Christopher Olah, `colah.github.io`)

RNNs are able to model temporal dynamics, such as a sequence of continuous joint configurations, where each frame is constrained by its prior. RNNs have therefore been popular in motion modelling.

1.2.2 Generative Adversarial Networks

Generative adversarial networks (GANs) are a method of training a network rather than a network type. In a GAN setting, one model, the generator produces an output, as normal. Instead of computing an error measure of the numerical distance between model output and ground truth, a second network instead decides whether the output appears "real". This second network alternatingly receives generator output and real motion and is trained using binary cross-entropy as a measure of how well it can discriminate the two; the discriminator is essentially a classifier. Instead of explicitly minimizing the distance between output and ground truth, the generator now optimizes its ability to fool the discriminating network – the two networks hence engage in a kind of minimax game where the generator tries to maximise, and the discriminator minimize the output of the following loss function:

$$L = E_x[logD(x)] + E_z[log(1 - D(G(z)))], \tag{1.1}$$

where $D(x)$ is the discriminator's estimate of the likelihood that an input sample x is real, E_x is the expected value over all true samples, $G(z)$ is the generator's output given input z (normally a noise vector), $D(G(z))$ is discriminator's estimate that sample produces by the generator is real, and $E(z)$ is the expected value over all inputs to the generator. This training process is illustrated in Figure 1.2. The generator and

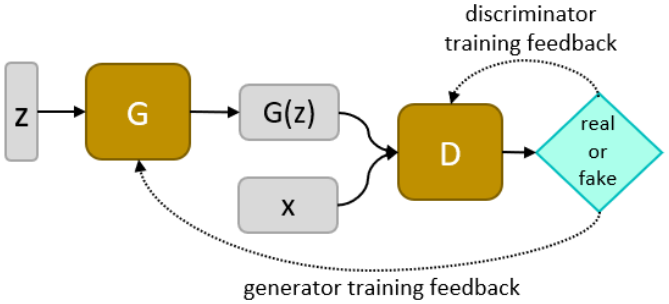


FIGURE 1.2: Schema of a generative adversarial network (GAN). The generator G receives an input z such as a noise vector and produces an output G(z). The discriminator D alternatingly receives this generator output and a real sample x and decides if a given sample is real or fake. The generator receives this decision as training feedback. The discriminator's training feedback is whether it was correct or not.

discriminator can have an arbitrary network architecture, for example, they can each be an RNN.

1.2.3 Transfer learning

Transfer learning is a method of applying knowledge gained from learning one task to a new task. For example, a model trained to detect cats in images may be re-used for a model detecting birds. The motivation is that the cat model already knows useful things for the bird task, such as segmenting images and detecting shapes. Another application is creating a specialized model from a general model, such as re-using the cat model to detect tabby cats. This can be useful when e.g. a large dataset is available for detecting cats, but only a small dataset for tabby cats; the general dataset can be used for initial model training and the specialized set for fine-tuning.

In practice, knowledge transfer between models is simply achieved by initializing the weights of the new model to be trained with the weights of the first model. Options include using only the weights of parts of the model, such as the first layers that represent more general feature extraction, as well as only allowing parts of the new model's weight set to be updated during training.

Transfer learning gives some constraints with regard to the architecture to the new model, as the origin layer dimension of the weights to be transferred needs to fit the target layer. When training the new model, the learning rate is usually initialized lower than for original training to avoid losing the transferred knowledge through large initial weight updates.

1.2.4 Performance measures

During the training phase of a machine-learned model, it is necessary to find numeric measures of performance in order to optimize the model. We make use of the mean squared error (MSE), a standard measure in machine learning:

$$MSE(p,t) = \frac{1}{n}\sum_{i=1}^{n}(p_i - t_i)^2, \tag{1.2}$$

where p is the predicted value and t is the true value.

However, due to the non-deterministic and highly variable relationship between speech and gesture expression, the MSE is not a sufficient measure of the quality of a model's output. A produced gesture may look plausible even if numerically far from the ground truth motion sequence.

To assess the performance of a gesture generation model, it is therefore important to include subjective measures, such as asking users how well the generated gestures match the concurrent speech. We included a number of such studies and utilized Likert scales for users' ratings. A Likert scale is usually a 5- or 7-point scale measuring user's attitude in intervals, ranging from one extreme to the opposite. In generative adversarial training (Section 1.2.2), the discriminator network mimicks a user's judgement of a generative model's output and replaces the need for an explicit error measure such as MSE.

1.3 Scope

In this work, we focus on machine learning methods for modelling the the relationship between speech and gesture motion. Many types of input can be used for designing a speech to gesture system, from speech recordings, to semantic speech annotation, as well as character gender, personality and mood. We focus on speech prosody as input for a gesture generation system due to its automatic extractability from audio recordings. We constrain our work to offline gesture generation rather than real-time; gesture naturally precedes or co-occurs with speech and it is therefore difficult or impossible to match gesture expression adequately to speech in real-time synthesis.

Contexts affect speech and gesture expression: Lecture-style speech and gesture differs from conversational, spontaneous speech. Here, we focus on spontaneous, unscripted speech in monologue-style. We focus on modelling a single actor speaking rather than including the dynamics of multiple speakers engaged in a conversation. We therefore do not address turn-taking or listening motions. To reduce the complexity of modelling gesture motion, we restrict our efforts to motion of the arm and hand joints. (In Chapter 5, we also include the spinal joints.)

No multimodal dataset of speech and 3D gesture motion of significant size was available at the outset of this work. We therefore recorded two large databases of conversational speech and high-quality motion.

1.4 Contributions

In a series of studies, we designed and assessed machine learning methods for modelling the speech-to-gesture relationship. We implemented a number of novel methods for this task and discuss the benefit and shortcomings of each. We first determined an inadequacy of standard regression training for our problem. We find that the highly varied and non-deterministic nature of the speech-gesture relationship may not be captured by a regression loss and lead to a minimizing of errors across all possibilities, namely mean pose convergence. We then addressed the problem of mean pose convergence by proposing an adversarial training targeted to assessing gesture motion characteristics. For this purpose, we designed a novel training of multiple objectives characterizing realistic gesture motion, one of our main contributions. As part of this novel motion assessment, we implemented and trained a network for automatically segmenting motion into gesture phases, allowing us judge gesture dynamics explicitly during training. Specifically, the phase network can detect if a motion is in a meaningful, expressive phase, in the process of preparation or retraction, or in a hold period. While we find adversarial training superior to a standard regression loss, we especially see promise in this phase separation of gesture.

Our next main contribution was a thorough investigation on how the characteristics of individual gestures during the expressive stroke phase relate to the concurrent speech signal. For this, we determined five gesture parameters and asserted their perceptual

importance for speech-gesture match. We design and implement a method to estimate these parameters from speech alone. With this, we were able to model expression of individual gestures from speech rather than continuous motion as most common for machine learning approaches.

Using the insights we gained about how gesture expression matches speech, we built a novel gesture generation system merging the use of machine-learned speech-gesture mapping and direct database sampling. Namely, we took speech audio as input and found matching gestures within a large database of motion-captured gestures, hence always producing natural and defined gesture form.

This database of gesture motion is furthermore a contribution of this work: We contribute two large multimodal datasets of speech recordings with synchronized high-quality motion-capture data, rich in gesture motion. Together, these datasets encompass over 10 hours of data, making this the largest open-source dataset of 3D motion and speech. This data has already had significant impact on the speech-gesture generation research community.

1.5 Summary of Chapters

The rest of this work has been divided into the following chapters:

- **Chapter 2** presents an introduction to gesture research as well as an overview over gesture generation methods.
- **Chapter 3** presents our two multimodal datasets of speech gesture.
- **Chapter 4** presents our investigation of the benefits of representing gesture motion in a lower dimensional space as well as employing a language model to address the complexity of the speech signal.
- **Chapter 5** presents a generative adversarial model of gesture generation, addressing the non-deterministic relationship between the speech and gesture channel that the classic training paradigm using a regression loss fails to capture. A split into multiple training objectives is proposed, phrasing the problem of appropriate gesture generation as a series of smaller sub-problems, including plausible gesture

dynamics and smooth motion. We present a method for automatically segmenting gesture motion into its dynamic phases.

- **Chapter 6** presents our study of which expressive aspects of a gesture may be modelled from speech. A perceptual study is presented on the impact of expressive parameters such as arm swivel and gesture velocity on speech-gesture match, and a series of machine learned models are trained to predict these gesture parameters from speech. We assess how well a particular parameter may be inferred from speech.

- **Chapter 7** presents a novel gesture generation system relying on the estimation of gesture parameters from speech by selecting a suitable gesture from the large database of gestures we built.

- **Chapter 8** summarizes and discusses the contributions of this work.

- **Chapter 9** explores future research in the area of co-speech gesture generation.

Chapter 2

Related Work

In this chapter, we will first briefly review research in character animation before diving into the intricacies of gesture motion specifically, from its definition to modelling.

2.1 Animating virtual characters

Animating a virtual character is, literally, bringing it to life. Movement almost immediately elicits perceptions of agency, as illustrated by such early studies as Heider and Simmel [16], who with their famous animation of an interaction between simple geometric forms (see Figure 2.1) showed, and indeed continue to show, how bringing movement to objects sparks a kind of story-telling in our heads. Animated movies as well as interactive animation in the form of video games consequently are a success story and much research surrounds the continuous improvement of animation technology. Interactive media in particular motivates the advances in motion production to enable realistic character control.

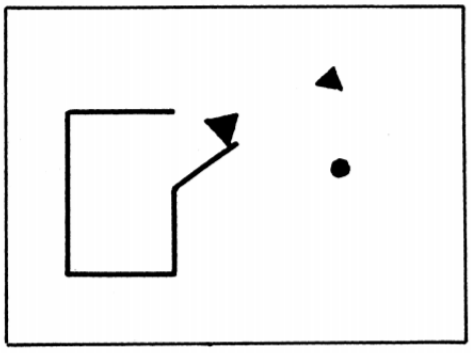


FIG. 1. EXPOSURE-OBJECTS DISPLAYED IN VARIOUS POSITIONS AND CONFIGURATIONS FROM THE MOVING FILM.
Large triangle, small triangle, disc and house.

FIGURE 2.1: Illustration from the historic study on motion perception by Heider and Simmel [16].

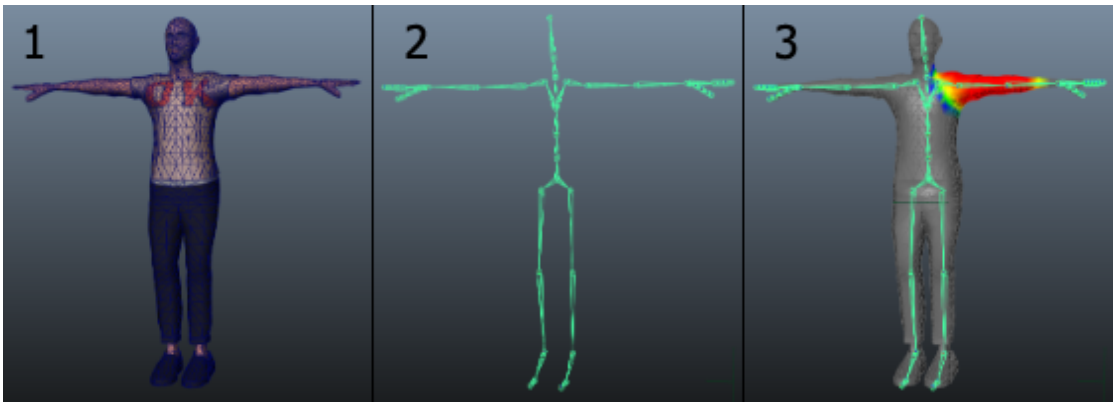


FIGURE 2.2: Steps of character creation. (1) Modelling the character, (2) Creating a rig for the character, and (3) fitting the rig to the vertices of the model (skinning). (Image: ©2018 Unity Technologies)

2.1.1 Animation representation

To animate a 3D model of a character (Figure 2.2(1)), a controllable rig needs to be created for it, for humanoid models this is commonly a skeletal structure (Figure 2.2(2)). Next, each joint in the skeleton is mapped to a number of vertices of the model, so that moving a joint will move the associated vertices. This is the process of skinning (Figure 2.2(3)).

Animations of the character are represented as sequences of joint rotations or positions, with local or global joint transformations. When using local transformations, the character's skeleton represents a hierarchy of joints, where moving one joint also moves its child joints that are lower in the hierarchy. For example, moving the shoulder joint will also move elbow, wrist, and fingers. In the case of global transformations, a joint's rotation or position is relative to an external defined coordinate system rather than relative to its parent.

Rotational representations are most commonly used in animation, specifically local rotation systems. A local rotation system allows for restricting joints to natural constraints through defining degrees of freedom (DOF). For example, hinge joints such as the knee or elbow can naturally only rotate around one axis with respect to its parent joint (1 DOF), whereas the wrist or shoulder can rotate around 3 axes (3 DOF). Rotations can be represented in a number of different ways. Euler angles are are 3 dimensional representation of a joint rotation (3 DOF), defined by sequentially applying rotations around the x, y, and z axis. The order of applying the axis rotations matter and need to remain

constant within one animation, but the choice of order can vary. An advantage of Euler angles is their intuitive format for a human reader. A disadvantage of Euler angles for computational models are discontinuous values: a rotation of 359 degrees is numerically far, but perceptually almost identical, to a 0 degree rotation. Furthermore, Euler angles can produce a so-called Gimbal lock, a loss of 1 degree of freedom, when two of the three axes are brought into a parallel configuration. As an alternative to Euler angles, quaternion representation is popular in animation. Quaternions are 4-dimensional vectors that have a less intuitive interpretation but do not suffer from Gimbal lock. One disadvantage is that a single quaternion cannot describe a rotation exceeding 180 degrees in any direction. The exponential map format is also relatively popular in computer animation and represents rotations by a unit vector describing the direction of an axis of rotation, plus an angle describing the magnitude of the rotation.

Position representations of joints are less popular in animation. Bone length constancy can more easily be violated as they are implicitly given through position differences, whereas they are explicitly defined for rotation representations. However, an advantage of positional representation, specifically global positional representation for motion modelling can be the fact that large movements, such as the hand tracing a big arc, are clearly marked by large numerical differences (e.g. the wrist position values changing significantly), rather than given through a number of inter-dependent joint rotations. This also allows for isolated joint analysis, such as the described hand trajectory, through just one joint transform rather than including all relevant parent joints (such as elbow and shoulder).

2.1.2 Motion capture

While animated movies still largely rely on hand animation by skilled animators, interactive media such as video games increasingly rely on motion capture for producing animations. Motion capture technology yields highly realistic motion, suitable for increasingly realistic game characters, without the need for hours of hand labour by animators. There are two major systems of motion capture. Optical systems use markers reflecting light generated by sets of special cameras to estimate body pose (see Figure

2.3). Inertial motion tracking systems use sensors directly on the body without external devices; body pose is estimated through data from accelerometers, gyroscopes, and magnets (see Figure 2.4).

Current motion research relies heavily on open-source datasets of motion capture. Some significant resources are listed in Table 2.1.

FIGURE 2.3: Left: Actress Ellen Page performing while her motion is recorded through optical an motion capture system. Right: The character animated with the captured motion in the game *Beyond Two Souls*. (Image: Quantic Dream)

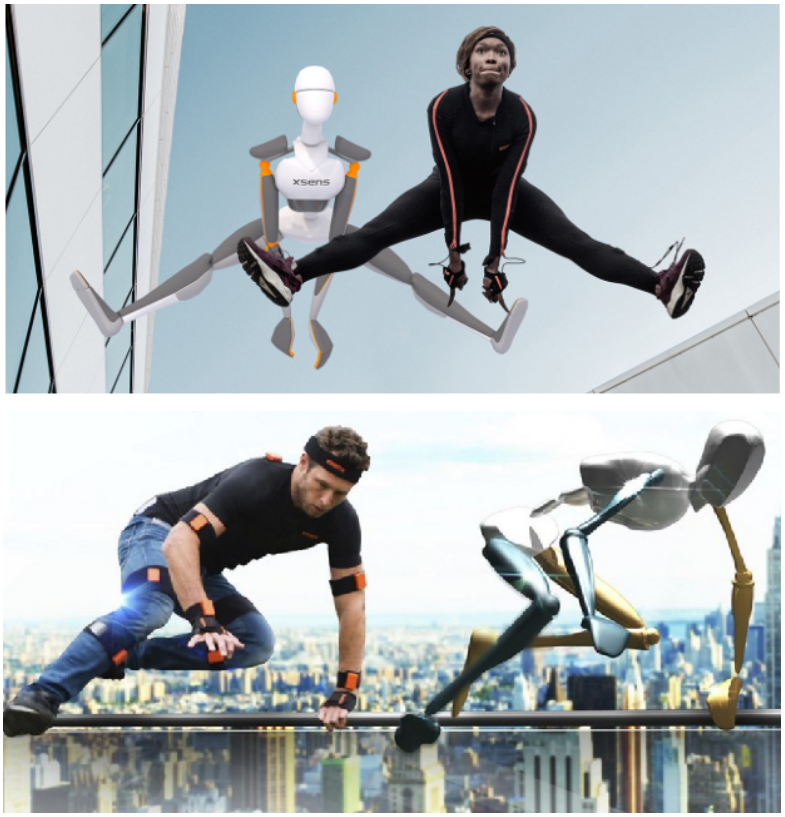


FIGURE 2.4: Inertial motion capture systems. Top: Sensors are integrated into a suit. Bottom: Individual sensors are attached to the performer.

TABLE 2.1: Major motion capture databases.

	CMU [17]	Human3.6M [18]	Panoptic [19]	Talking With Hands 16.2M
hours	-	20	5.5	20
sequences	2605	-	65	116
motions	locomotion, dancing, interaction, ...	discussion, talking on the phone, taking photos, eating,...	conversation, dance, musical performance,...	conversation

2.1.3 Controllable character animation

Interactive control of animated characters relies on transitioning between predefined animations (often snippets of motion capture) based on user input, for example, the character should transition from walking to a running animation, and then jump over an obstacle while running. To define transitions between animations, a common approach is the use of state graphs, also called animation state machines, defining actions as states and connections between states representing transition times. Authoring a state machine quickly becomes tedious when large amounts of different actions should be combined into the graph. Borer et al. [20] proposed a method to partially automate the creation process of such a state machine for a controllable character. Using temporal replanning, all desired actions of an agent can be merged into a behavior plan. For example, if two desired actions overlap, temporal replanning takes timing and priority of the actions into account to prematurely end the first action, or finish the first action before creating a transition to the next.

Motion graphs represent another method of automatically creating suitable connections between motion segment [21]. Motion graphs are directed graphs consisting of pieces of captured motion and synthetic transitions between them. A coherent motion sequence can be generated by taking a path through connected states in the graph.

Avoiding the creation of a state graph entirely, the Motion Matching method proposed by Clavet and Büttner [22] draws animations from a database based on a set of specified motion properties and combines them with simple blending and inverse kinematics. Animations are selected from the database based on the action to perform by the character, as well as motion parameters such positions of the end effectors and the past and present

trajectory. The method became popular with gaming studios due to its reliable motion quality and its suitability for real-time animation through the use of efficient nearest neighbor search.

A limitation for both Motion Graphs and Motion Matching can be memory usage for larger datasets, increasingly available through widespread use of motion capture technology. To address this concern, Holden et al. [23] proposed a combination of Motion Matching and neural-network based controllers, which includes compressing motion data to a low-dimensional representation, reducing memory usage significantly. Increase in motion data availability, however, has also enabled an entirely different approach to character animation, namely the automatic generation of the motion itself rather than just the transitions between predefined animation snippets.

2.1.4 Learned character animation

Locomotion such as walking and running is arguably the most successfully modelled aspect of human motion, providing a relatively constrained objective due to its periodicity and relative uniformity across subjects and time. By learning a low-dimensional manifold of locomotion data using a convolutional autoencoder, Holden et al. [24] proposed an offline framework for motion generation taking into account user input, adapting to terrain surface, and allowing for style edits. Following this, Holden et al. [25] harnessed the cyclical nature of locomotion using a Phase-Functioned Neural Network, enabling real-time biped locomotion across rough terrain, including jumping and avoiding obstacles. Lee et al. [26] proposed the use of Recurrent Neural Networks utilizing motion graphs to train a memory- and computation-efficient control network that simulates graph-based motion authoring. Henter et al. [27] argued for a probabilistic model using Normalizing Flows, naming the advantages of being able to produce realistic motion even with a weak, under-constrained control signal and for non-periodic, varied motions in order to generate varied motions, responding to control inputs with zero latency.

The extensive research in motion generation and control provides important guidelines for our task at hand, namely producing body motion specifically for a speaking agent. However, this task also has two important differences with respect to the work above: Firstly, the input control signal is very weak; instead of a specific signal such as “move the arm in a circle”, the signal is represented by the co-occurring speech (e.g. “and she

went around and around"). Secondly, gesture motion is highly varied and unconstrained; whereas e.g. a walking motion looks relatively similar across people and relies on the feet periodically making contact with the ground, gesture motion varies hugely between people, in style, in shape, in frequency, and more.

2.2 Defining gesture motion

For our aim of modelling and generating gestures, we firstly want to understand the concept of a gesture. Which movement can be classified as a gesture? Here, the literature proposes some differing definitions. Cassell [28] suggests defining gesture loosely as a "motion of the limbs or body made to express or help express thought or to emphasize speech", not, however, providing observable motion characteristics for analysis. McNeill [15] proposes a more strict definition, classifying movement of the arms and hands as a gesture when it is "closely synchronized with the flow of speech", implying that gesture motion cannot stand alone, without speech. This postulation was partly based on the observation that listeners in the author's many hours of recordings did not produce gestures aside from a single instance. This may also give insight to the *function* of gesture, discussed below. McNeill [15] also states that the emphatic core of the gesture motion (the gesture "stroke", further discussed in Section 2.5) precedes or coincides with the prosodic peak of the speech, but does not follow it, and bases this on findings of Kendon [29]. Nobe [30] find that in about 90% of cases, the gesture precedes the respective speech. Further evidence for these tight links between speech and gesture production was found by studying clinical stuttering; Mayberry and Jaques [31] found that onset of stuttering causes immediate abortion of the gesture stroke, and onset of a gesture stroke inhibits stuttering.

Semantically, McNeill [15] asserts that co-occuring speech and gesture convey the same underlying idea. This does not, namely, imply gesture to provide redundant information, but rather complementing and adding to the speech information. This pragmatic aspect of gesture relates to its function, offering the listener a better understanding of the portrayed sentiment.

2.3 The function of gesture

Evidence for the benefit of co-speech gesture to the listener's understanding was provided by Cassell et al. [32], who found that when retelling a narrative, listeners were able to describe information that was only portrayed in gesture, not speech. On the other hand, mismatches of speech and gesture, that is, when the information provided by the gesture did not match the information provided by the accompanying speech, were found to significantly increase retelling inaccuracies. Adaptive teacher gestures can help children learn [33], while artificially delaying gesture motion detrimentally affects learning in children [34]. Even non-meaningful gestures shape the way we perceive speech: Bosker and Peeters [35] report a kind of manual McGurk effect, where rhythmic gestures influence which vowels are perceived by listeners through modulating perceptions of lexical stress.

However, the influential role of gesture may not be restricted to the listener. Rimé et al. [36] restricting subjects' use of gestures also restricted subjects' verbal expressiveness. Restriction of gesture motion also elicited increased motor activity in eyebrows, eyes, and fingers, areas the authors also identified as being associated with verbal processing. After further analysis of the produced speech, the authors noted an increased amount of words used by the speakers under gesture restriction, while simultaneously expressing themselves less clearly and with less fluidity [37]. Indeed some researchers of language express serious doubt about the benefit of gesture for the *listener*, observing that the vast majority of gestures in their experiments were produced from the speaker's perspective [38], and that gestures offer little useful information for the listener [39, 40], do not significantly alter the interpretation of the speech content [40], and do not aid the listener's verbal understanding [41]. Focusing instead on the facilitatory role of gesture for the *speaker*, Krauss and Hadar [38] report a link between gesture and lexical memory. Their experiments showed that restricting the speaker's gesture behavior led to difficulty in word retrieval. Goldin-Meadow and Wagner [33] further showed that restricting gesture negatively affected memory in a learning task. Pouw et al. [42] found another benefit of gesture motion for the speaker, namely aiding speech vocalization by modulating pitch and intensity. Additional evidence for the speaker-centric role of gesture comes from findings that even children that are blind from birth produce gestures, and their

gestures resemble those produced by sighted children, and this is true even when communicating with a known blind listener [43]. Gesture may therefore be a natural part of speaking without serving a communicative intent for the listener. Note, however, that by facilitating verbal expression for the *speaker*, gesture nonetheless indirectly benefits the *listener* through the improvement of speaker fluidity and verbal expressiveness. A potential reason for the different conclusions regarding the function of gesture drawn by different researchers may be that there are different types of gestures that may fulfill different objectives.

2.4 Types of gestures

Gestures are usually classified by the four categories proposed by McNeill [15], into iconic, metaphoric, deictic, and beat gestures. Iconic gestures visualize physical properties, describing the semantic content of the verbalisation. For example, the speaker may move his or her hands down, with flat, open palms, while saying, "he was pressing it down". An iconic gesture can also add information to the verbalisation, as in the example provided by McNeill [15], where the utterance "and she [chases him out again]" is accompanied by the hand appearing to swing an object through the air. The verbalisation names the action performed, whereas the gesture suggests the manner in which the action was performed. Both channels of communication add to the understanding of the underlying idea or thought of the speaker.

Metaphoric gestures portray an abstract idea rather than a literal physical description. For example, consider the above used gesture description of both hands moving down, with a flat, open palms, accompanied by the phrase, "they are suppressing women". Here, the speaker likely does not mean that the women were physically pushed down, but rather is associating the concept of women's stifled role in a matter with pushing something down.

Deictic gestures are pointing gestures, such as pointing at an object while saying, "it is over there". However, the object indicated with the gesture may not be physically present but rather have an implied presence through the narrative and the building of the gesture space.

Beat gestures are gestures that do not portray any specific meaning. They are, however, closely linked to the rhythm and pace of the speech [44]. They can serve to emphasize a verbalisation and are often co-occurring with stressed words or syllables. They can range from small flicks of the hands or fingers to large arm motions. In conversational discourse, beat gestures have been reported to make up the majority of gestures [45–47].

2.5 Structure of a gesture

Through further analysis of a gesture, it can be segmented into phases with qualitatively different dynamic characteristics [29] occurring in specific patterns [48].

2.5.1 Gesture phases

First, in the *preparation* phase, the hands are moved into position for the gesture to be performed. Next follow the core, meaning-carrying movement of the gesture, the *stroke*. It is the expressive phase of a gesture and has the most focused energy, described as an "accented movement" with Effort in the sense of Laban [48] (see Section 2.6.1), conveying a sense of intention and meaning of the motion. In the case of iconic gestures, this is the phase describing a specific shape that relates to the accompanying verbal phrase [15]. The *retraction* moves the limbs back into a restful position. Sometimes the hands are only partially moved back towards a rest position, before continuing to the next preparation or stroke; this incomplete retraction is noted as a *partial retraction*. *Holds* are segments with zero velocity and may occur before (pre-stroke hold) or after the stroke (post-stroke hold) [49]. (An example of such a sequence of phases within a gesture is shown in Figure 2.5.) Pre-hold strokes are thought to serve as a moment for the speech to catch up to gesture, so that the gesture can be performed in synchrony with

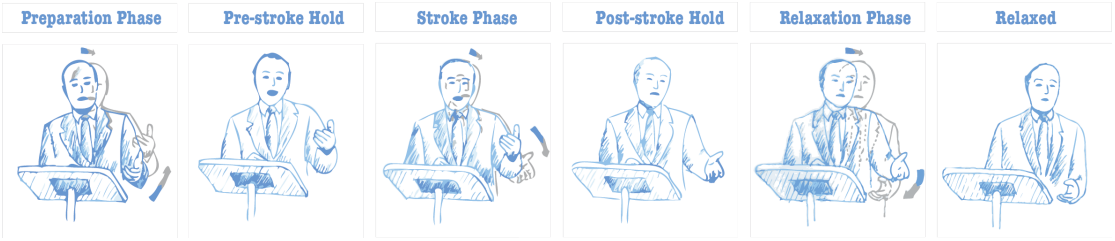


FIGURE 2.5: Example phase sequence of a gesture. (Image by Ada Ren, `http://web.mit.edu/pelire/www/gesture-research/index.html`.

the associated speech part [49]. Post-stroke holds may be a way to extend the temporal duration of a gesture, so that the stroke together with the hold are synchronous with the associated speech part [50]. Another explanation for the function of post-stroke holds was put forward in Duncan [51], observing that the hold may express a prolonged state of an idea. A third type of hold can replace the stroke phase entirely, this is a so-called *independent hold*, existing independently of a stroke. An independent hold describes the meaning of the gesture by its shape, for example by describing a sign, often specific to a cultural region, such as the thumbs-up sign or the peace sign, or to describe enumeration (showing "one time" by lifting the index finger).

The stroke phase is the only essential part of a gesture (but can be replaced by an independent hold), whereas all other phases are optional. That is, a gesture (stroke) may not have a preparation but rather continue immediately from the previous position, and similarly, instead of being followed by a hold or a retraction, can be immediately followed by another gesture. A sequence of gestures that ends with a retraction to a rest pose is also called a gesture unit [52]. As the stroke phase contains the gesture form and contains the gesture's meaning (or, in the case of beat gestures, the emphasis), we may be most interested in separating the stroke phase from the general motion, enabling analysis of an individual gesture.

2.5.2 Segmenting gesture into phases

Segmenting gesture motion into its phases is non-trivial and in many cases requires subjective judgment. Hence the labelling process cannot be seen as deterministic and 100% accuracy is unlikely, or even impossible. Often, gesture phases can be straightforward to identify, but in other cases, it may be more difficult. This tends to occur when one stroke goes directly into another or if a stroke starts from a retract position. Consider for example the ambiguous example of a gesture sequence in Figure 2.6, where both step (1) and (3) are determined to be a stroke phase: One could consider the motion to the middle transition frame (2) either a partial-retract of the first stroke in (1) or a preparation for the second stroke in (3).

The work of segmenting gesture recordings into phases is tedious; segmenting just one minute of video into gesture phases may take one hour or more of work (e.g. [2]). Different, automatic gesture phase annotation methods have hence been proposed, including

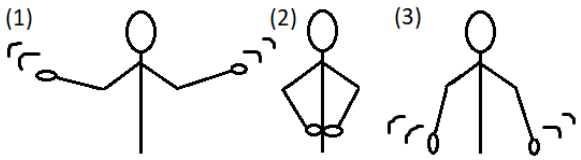


FIGURE 2.6: Ambiguity in gesture sequence labelling. If steps (1) and (3) are each considered a gesture stroke, the motion to the transition step (2) may be labelled as either a partial-retract of the preceding stroke or a preparation phase for the following stroke.

the use of support vector machines [53] and hidden Markov models [54, 55]. One limiting factor in training phase models is obtaining labelled data, which, again, takes many hours of skilled work. Previous work has therefore often focused on simpler sub-problems of detecting whether one specific phase is occurring (e.g. detection only of gesture strokes), or whether a gesture is being performed at all. Bryll et al. [56] use heuristic classifiers to detect holds from video. Gebre et al. [57] detect gesture strokes from video using a linear classifier. Alexanderson et al. [55] use hierarchical HMMs to extract gesture boundaries, resulting in a segmentation into rest, gesture, and manipulator (e.g. touching of one's own face or hair).

Another difficulty in automatic phase detection is the difference in phase structure as well as phase expression between speakers and even within speaker. Phase structure differences can include overall gesture rate as well as differences in the distribution of phases; for example, one speaker may regularly produce two or more gesture strokes before returning to a rest position, while another speaker may average just one stroke before returning to rest [58]. Phase expression such as the stroke velocity profile can vary not only from speaker to speaker, but also between recordings of the same speaker [53]. This also means that detecting and segmenting stroke phases may enable easier speaker comparisons, allowing comparisons of stroke length, speed, frequency, etc.

2.6 Gesture expression

Gestures differ not only by their form but also in the way they are performed by the speaker. The same gesture may be expressed in an empathic, energetic manner, or with a sluggish motion. Gesture expression can vary both with the mental state of the speaker, as well as between speakers, who each may have their own personal way of performing gestures. Several works have looked at ways to analyze and describe the

movement characteristics of gestures during the stroke phase. By finding measurements of gesture expression, we can investigate how gesture expression relates to the speaker's mental state, both temporary and permanent, specifically their emotional state and personality.

2.6.1 Laban Movement Analysis for gesture

Laban Movement Analysis, a framework for systematic description and evaluation of human motion has been employed for this purpose, specifically its Effort and Shape parameters. Bartenieff and Lewis [60] describes gesture as "any movement of any body part in which Effort or Shape elements or combinations can be observed".

Effort hereby describes the dynamic quality of the movement rather than its content, describing the energy used for the motion and its rhythm and timing. North [61] argues that Effort is unique to a person, describing an individual's way of moving. Effort consists of four factors, described by Laban and Ullmann [62] and visualized in Figure 2.7: (1) Space (ranging from Direct to Indirect), describing whether the motion follows a direct, straight trajectory, or a wavy, flexible trajectory. (2) Weight (from Strong to Light), describing a motion as ranging from heavy with a feeling of resistance, to light, with a feeling of gentleness and weightlessness. (3) Time (from Sudden to Sustained), describing whether a motion is quick and momentary, or slower, giving a feeling of long extend through time. (4) Flow (from Bound to Free) underlies all movement expressions and describes how controlled versus released the motion is, ranging from hampered motion to fluid, free motion.

The Shape dimension of Laban described the shapes and their changes described by a moving body. Shape consists of several sub-categories, with the perhaps most relevant to gesture motion being the Modes of Shape Change, describing the bodies interaction with

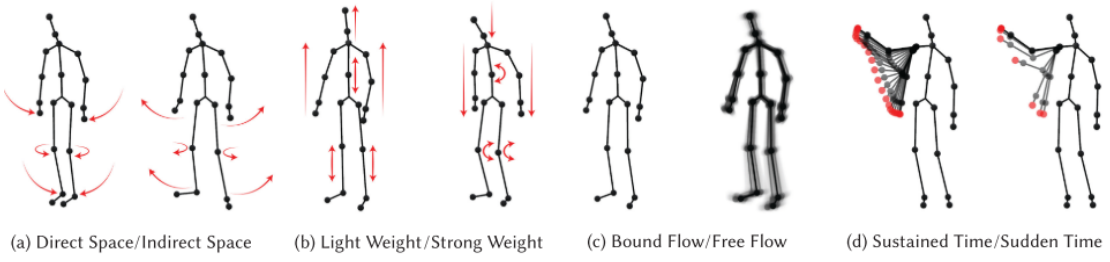


FIGURE 2.7: The four Laban Effort factors. (Image from Sonlu et al. [59])

and relationship to itself and its environment through three factors: (1) Shape Flow, describing the bodies relationship to itself, or its body parts' relationship to each other. (2) Direction describes the bodies directional relationship to a part of the environment, such as reaching towards an object. (3) Shaping, describing motions such as tracing the shape of an object with the hands. In addition to Shape Change, Shape has a category of Form, the body expressing a static shape, and Shape Quality, whether the body movement is Opening (the limbs expanding outwards) or Closing (moving toward the body center).

In Section 2.7, we will discuss further how Laban's Effort and Shape parameters have been employed in the domain of gesture animation.

2.6.2 Low-level motion parameters for gesture

Other motion parameters have been proposed to describe and analyze motion specific to co-speech gesture. One motivation for assessing different parameter representations for modelling gesture is the difficulty in obtaining and manipulating Laban parameters, requiring many hours of trained experts' work. Alternatively, more easily extractable motion parameters have been proposed to describe gesture. A plethora of measurements is available here; previous work has often grounded the selection on social psychology literature of bodily expressions of emotion and personality, and we will discuss these findings in the next Section. Easily obtainable measurements are for example the handedness of the gesture (is the right hand or the left hand performing the gesture, or both?), the palm orientation and shape, the height of a gesture and the direction of the gesture motion. If motion capture data is available, fully automatic methods can also easily extract quantitative descriptors such as gesture velocity and acceleration.

2.6.3 Emotion in gesture

The study of bodily expressions of emotion dates back as early as Darwin [63] with his work on "The expression of the emotions in man and animals", who for example names pushing away gestures as signs of disgust. Ekman and Friesen [64] later doubt the existence of any specific gesture, movement, or posture for a specific emotion, but rather emphasizes the importance of the quality of the movement (compare to Laban's

Effort versus Shape, Section 2.6.1). Camras et al. [65] find evidence for qualitative motion differences between emotions, reporting for example that anger elicited more jerky and active motion than sadness, later attributed to some extent to the dimension of activation by Wallbott [66], with anger representing an active, and sadness a passive emotion.

A number of works have employed the Laban movement parameters to assess expressed emotions. Levy and Duke [67] videotaped participants in a guided movement improvisation task and subsequently assessed their emotional state through depression and anxiety scores. The authors found a number of associations between the Laban movement parameters and mental state. Noting a difference for genders, the authors further report male subject with higher anxiety to produce more enclosing movements, whereas female subjects with high anxiety showed decreased use of sagittal movements. Both males and females with higher scores for depression also showed less sagittal movements, depressed males additionally showed an affinity for Indirect Space, and depressed females a decreased changing between efforts.

Expressly manipulating their emotional state, Morita et al. [68] investigated participants' more short-term disposition. In this study, participants first listened either to pleasant or unpleasant sounds while their movements were recorded on video. Then, participants self-reported their emotions. Laban parameters were estimated from the video recordings, and the authors found significant differences of Laban expression between participants experiencing positive versus negative emotions. For example, reported anger and fatigue were positively correlated with Laban's Weight component, indicating more active movement, and tension increased Laban's Time Space measures, indicating more hurried movement changes. Truong et al. [69] used Laban measurement to judge emotion in orchestra conductors' movements with some success. Nakata et al. [70] developed mathematical definitions of Laban parameters and use them to create dancing motion for simple robot. They find that, for example, Weight Effort Strong was perceived as joyous motion, whereas Weight Effort Light was often perceived as sad, and Advancing motion tended to be perceived as angry. Masuda et al. [71] use a more complex humanoid robot and propose another estimation algorithm mapping Laban's parameters to four emotions, achieving high emotion recognition scores for the robot motion.

Moving away from Laban representation, Volkova et al. [72] assessed subjects' ability to detect emotion from upper body motion and identify motion parameters such as speed and the span between the wrists to be associated with specific perceived emotions. For example, they found motion with wider wrist spans to be more often perceived as joyous or surprised than as fearful or ashamed. By studying recordings of theater performances of two actors, Kipp and Martin [73] found right-handed gestures to be associated of performance of negative and aggressive emotions, and left-handed gestures being performed more often to portray positive and relaxed feelings. However, as the authors rely on staged performances and only two subjects for their analysis, it is unclear how generalizable their findings are. Indeed, Castillo and Neff [74] later reports opposite findings for handedness. Here, the authors investigated the perception of emotion through gesture by systematically manipulating gesture performances with 11 modification parameters. Using Russell's Circumplex model of emotion, they could significantly change perceptions of the Valence and Arousal dimensions. For example, increased tension of the motion and left-handed gesture were perceived as having lower valence, and higher as well as longer gestures were perceived as having higher arousal.

2.6.4 Personality in gesture

Another line of work has investigated correlation of gesture expression with stable personality traits. Gallaher [75] report some correlations of four content personality traits, anger, fear, activity, and sociability, with four dimensions of movement style, Expressiveness, Animation, Expansiveness, and Coordination. For example, they find Expressiveness to correlate with sociability and Expansiveness with fear.

Levy and Duke [67] measures personality correlates of Laban movement parameters, finding different patterns for males and females. Males with high achievement scores tended to produce less enclosing movements, but no such pattern was found for females. Dominant males also showed less enclosing movements, while dominant females showed the opposite trend, producing less spreading movements. While affiliation in males correlated positively Bound Flow and Direct Space movements, affiliative females preferred movements with Strong Weight and tended to change between Effort qualities.

More recent work utilizes the OCEAN model of personality, also named the Big-Five, to assess interpersonal differences in character. The OCEAN model encompasses five

factors of personality, Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism.

Extraversion in particular as the most outward-oriented, social dimension has been under frequent investigation for gesture behavior. Riggio and Friedman [76] found extraversion to be correlated with gestural fluency, and Lippa [77] reported extraverts to portray more energetic gestures, having the hands away from the body more often, producing faster speech and, with that, more frequent gestures, and exhibiting more arm swivel (larger distance between the elbows and the torso). Extraverts have further been found to produce more spatially expansive gestures [78] as well as producing more varied gestures [79]. Other work, however, has noted the importance of the personality of the *conversation partner* for the speaker's gesture style. Tolins et al. [80] matched and mis-matched conversation pairs of introverts and extraverts and found the conversers to adapt their gesture style to their partner over the course of the conversation. For example, introverts started with narrow gestures but, when conversing with extraverts, their gestures became broader over the course of the conversation. Extraverts increased their arm swivel when matched with another extravert, but decreased arm swivel over time when speaking to an introvert. The authors also reported an opposite relationship between extraversion and gesture rate, compared to previous research, reporting that introverts displayed higher gesture frequency.

Based on the findings of human social psychology studies, continuing research has employed virtual agents to systematically manipulate gesture styles, suggesting design methods for expressive virtual agents with personality. By modifying eight factors of

Fig. 2: Gesture performance extraversion styles in order of increasing extraversion

FIGURE 2.8: (Perceptual study by Neff et al. [12]. Gesture scale was adjusted to manipulate perceptions of extraversion.)

gesture performance, including, scale, arm swivel, position, and duration of the gesture, as well as modifying overall gesture rate, research has shown that perceived extraversion of an animated agent can be manipulated [12] (example in Figure 2.8). Smith and Neff [81] extended this work by using a set of parameter modifications to target perceptions of all Big-Five personality traits. Again, perceptions of extraversion was found to be most modifiable by adjusting motion parameters, positively correlating with increased gesture size and velocity, and a strong effect of finger extension, with extended fingers increasing perceived extraversion. Arm swivel, disfluent gestures, as well as clavicle lift increased perceived neuroticism and decreased agreeableness. Disfluent gestures and clavicle lift additionally decreased conscientiousness. Similar work modified only the amplitude and speed of a robot's gestures, finding higher gesture amplitude and speed to elicit higher rating of extraversion and neuroticism [82]. Wang et al. [83] focused solely on the effect of hand motion on perceived personality and found both hand pose and amplitude of motion to affect perceptions of all five personality traits. For example, spread fingers received high rating of extraversion, openness, and neuroticism, and low ratings of conscientiousness and agreeableness.

2.7 Modelling gesture motion

For creation of believable agents, previous works have stressed the importance of capturing emotion [84] and personality expressed through gesture [85]. To capture these expressive aspects, a number of computational models have been proposed for synthesizing new gesture motion for animated virtual agents

For animating expressive agents through descriptive Laban parameters, Laban descriptors have to first be mapped to lower level motion characteristics that can be used to modulate animations algorithmically. For the problem of mapping Laban parameters to easily quantifiable motion characteristics, research has often focused on dance movements [86–88], which may show Laban parameters more clearly and pronounced than natural gesture motion, but efforts have been made to extract Laban's parameters from general motion [89], and, importantly, gesture motion [90]. Here, the authors proposed an approach to extract Laban parameters from hand and arm movements using a predefined gesture repertoire of 6 motions, each performed by an actor in 6 emotional

variations. They quantify Laban parameters through motion analyses such as the average trajectory curvature. The authors report good results in extracting the Laban parameters of Time, Weight Effort, and Shape Directional, while Space and Flow were more difficult to capture.

Basing their model on the Effort and Shape parameters of Laban, Chi et al. [91] proposed a computational model for animating gesture motion. Their EMOTE (Expressive MOTionEngine) framework maps low-level motion attributes to Effort and Shape dimensions, allowing an animator to synthesize co-verbal gesture behavior by manipulating Effort and Shape. Zhao and Badler [92] extended the EMOTE system, making steps towards reversing the synthesis process to extract Effort from 3D motion. Durupinar et al. [93] further extended the EMOTE model by working together with a number of movement analysts to model personality-expressive motion by mathematically mapping between OCEAN personality traits, Effort, and low-level motion parameters. For example, they show that the Effort dimension of Time can be modified by animation speed, directness of trajectory, and breathing frequency, and animating sustained Effort increased perceptions of agreeableness while decreasing perceptions of extraversion. Sonlu et al. [59] provide a full framework for modulating perceived personality through manipulations of Laban's Effort and Shape, including methods of manipulating the Shape qualities Rising, Spreading, Advancing of gesture, as well as the Effort qualities of Time, Flow, Space, and Weight.

Hartmann et al. [94] proposed an alternative set of motion parameters to capture the expressivity of gestures. Based on a review of social psychology literature as well as analysis of a gesture corpus, the authors determined six motion parameters: (1) Overall Activation, the quantity of motion, (2) Spatial Extent of the motion, (3) Temporal Extent (quick versus sustained, similar to the Time factor of Laban's Effort), (4) Fluidity (smooth versus jerky, similar to the Flow factor of Effort), (5) Power (weak versus strong, comparable to the Weight factor of Effort), and (6) Repetition, the rhythmic repeats of motion. The authors propose a computational model to map these 6 parameters to modifiable animation parameters. For example, they modify the parameter of Spatial Extent by changing arm swivel by moving hand inverse kinematics targets outward. Some parameters, such as Overall Activation, were found to be difficult to capture with the proposed mapping. They report some evidence that matching expressive parameters to the communicative intent makes the gesture behavior more appealing.

2.8 Speech-driven gesture generation

To create gesture behavior matching the communicative intent of a virtual agent, a mapping has to be created between the two. While gesture behavior can be hand-designed to match a given utterance, this requires a skilled animator and significant amounts of time. To remove the need of hand-authoring gesture behavior for speech content, much work has investigated ways to produce gesture motion automatically from the accompanying speech signal. This takes the work presented in the previous section a step further - rather than only modelling the gesture motion space, the goal is to find a mapping from the speech signal to this motion space.

It is worth noting that the vast majority of gesture generation systems do not work in real-time. We have discussed that gesture either precedes or co-incides with the respective speech utterance in Section 2.2. To honor this constraint, a gesture generation system would have to plan and initiate a gesture before the related speech is produced, and can hence not base gesture shape or expression on this speech segment. However, Wang and Neff [95] found users could not detect gesture delays of up to 0.6 seconds (though in side-by-side comparison sensitivity increased to about a 0.2 second threshold), and Nirme et al. [96] similarly found that 0.5 second delays or advances went unnoticed unless gesture strokes overlapped with pauses in speech. This gives a small but potentially useful window for developing real-time systems.

The research into gesture generation from speech can largely be divided into three categories, namely rule-based systems, statistical modelling methods, and machine learning approaches.

2.8.1 Rule-based gesture generation

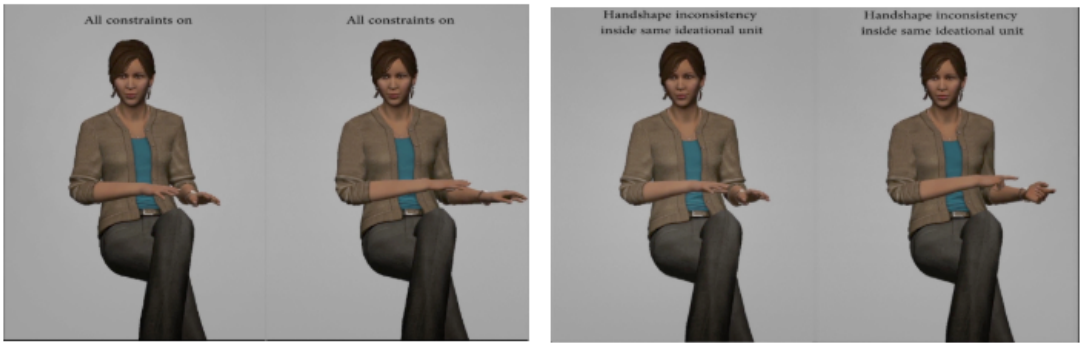
The first approaches to model the speech-gesture relationship have employed rule-based methods, explicitly formulating mappings from the speech to the gesture dimension. The Behavior Expression Animation Toolkit (BEAT) first proposed an automatic gesture generation system based on input text [97]. A linguistic processing module segments utterances into clauses, and clauses into themes (connecting a clause to the previous one) and rhemes (new information of the clause). Based on the finding that gestures generally coincide with a rheme [98], gestures are generated with in the following manner:

```
FOR each RHEME node in the tree
    IF the RHEME node contains at least
      one NEW node
    THEN Suggest a BEAT to coincide
      with the OBJECT phrase
```

The generator also determines if any named objects and actions can be found in the knowledge database. If a match is found, a meaningful gesture can be sent to the animation module. A similar system was proposed by Lee and Marsella [99], aiming to provide a clearer and more standardized interface by using FML and BML standards (functional and behavioral markup language, respectively). The presented system became an integral part of the open-source animation framework Smartbody [100].

A framework focusing on deictic (pointing) and iconic gestures (e.g. miming a spatial orientation, such as "crosswise") was proposed with Max, the "Multimodal Assembly eXpert" [101], designed for an interactive assembly game. Max can help assemble virtual structures by pointing at locations of parts and miming spatial relations through predefined gestures. Similarly, Kappagantula et al. [102] proposed a system for deictic gesture production of pedagogical agents by searching for mentions of element locations in the speech and applying inverse kinematics. In Marsella et al. [103], a new method of processing input speech was proposed, specifically, including acoustic features rather than relying solely on input text. By analyzing the prosody of input utterances, the system detects stressed words as well as the overall level of agitation of the speaker. A lexical analysis of the speech content extracts communicative functions such as affirmation or emphasis, and then maps these functions to a gesture set via 97 rules. Xu et al. [1] emphasize the relationship between multiple gestures, not just between speech and gesture, realizing the concept of ideational units proposed by Calbris [104]: Consistency of topic should produce consistency in motion aspects of gestures, and a shift in topic should be marked by a shift in gestural form and movement (see Figure 2.9).

Even metaphoric gestures have been addressed with rule-based approaches. Lhommet and Marsella [105] proposed a system specifically aimed at metaphoric gestures, using a set of rules mapping physical properties to metaphoric ideas. For example the word "anything" corresponds to the concept of a container containing everything, which in turn must be a big container, and hence "anything" is mapped to the physical property



(a) All constraints on **(b)** Inconsistent handshapes

Fig. 5. A comparison example: each image shows two frames of the performance. Left image has all constraints active and right image disables one constraint - the handshape consistency within the same ideational unit.

FIGURE 2.9: A rule-based system proposed by Xu et al. [1] honoring ideational units in generating gestures.

of a big container, eliciting a gesture shaping a big container. Ravenet et al. [106] developed this approach further, proposing a method to automatically extract various metaphorical physical properties from input speech, and mapping these to the gesture descriptors of hand shape, wrist position, palm orientation, and arm movement type.

An advantage of rule-based systems are that they do not require obtaining a large dataset of natural gesture, or even any at all. Gestures can be hand-defined by an animator, and mapped to the desired communicative function. Another advantage is the ease of generating meaningful, expressive gestures; as we will see in the following Sections, statistical models may need a dataset containing a number of examples of the desired meaningful phrase-gesture co-occurrence, and machine-learning models may still have a hard time capturing semantically meaningful gestures at all. However, while the above presented rule-based systems often designed the language-to-gesture mapping to be tuneable and extensible, their expressiveness is always limited by the number of hand-defined gesture correlates. Furthermore, the gesture behavior is based on the ideas of the rule creator rather than actual human gesturing behavior, unless rule creation is based on extensive study of human recordings. Note also the relation of phrase-to-gesture rules to the more out-dated idea of specific emotions being associated with specific gestures, described at the beginning of Section 2.6.2.

2.8.2 Statistical models for gesture generation

A second approach to gesture generation from speech are statistical models, addressing the wish to base gesture selection on actual human behavioral data. These works rely on hand-annotation of a corpus of recordings of human conversation, tagging gestures, or gestural features, as well as speech features.

Statistical modelling methods use estimated conditional probabilities of certain speech features co-occurring with specific motion features. For creating these statistical models, it was found to be beneficial to develop speaker-specific gesture production [107]. Neff et al. [2] developed a model to capture such speaker-specific gesture style. The authors annotated video corpora of the two talk show hosts Jay Leno and Marcel Reich-Ranicki with 30 different gestural lexemes, such as "Cup" or "Wipe", and about 90 different semantic tags, such as "number", "agreement", and "positive affect". A statistical gesture model is computed to map the semantic tags to the gesture lexemes; following this, new input text can automatically be processed and the gesture motion can be produced in either the style of Jay Leno or Marcel Reich-Ranicki (see Figure 2.10). Synthetic preparation and retraction motions for the generated gesture strokes are produced through simple pose blending, creating coherent gesture sequences. For small time windows between two strokes, pose blending can instead be applied between the end pose of one gesture and the start pose of the next gesture.

In a similar approach, the Gesture Net for Iconic Gestures (GNetIc) used Bayesian decision networks and hand-coded gestures with more detailed descriptors, including handshape, palm and finger orientation, and movement direction [108]. The model again computes the conditional likelihoods of gesture characteristics for linguistic features,

FIGURE 2.10: A gesture sequence of Marcel Reich-Ranicki re-created on a virtual character by Neff et al. [2].

specific to each one of five speakers. It was later extended into a cognitive behavioral model for concurrently planning and producing speech and co-speech gesture [109].

Fernández-Baena et al. [110] create a detailed annotation of 6 minutes of an actor performing beat gestures (gestures without a specific meaning) to analyze correlations of speech and gesture with respect to synchrony and intensity. A motion graph is built and used for gesture synthesis by searching the gesture with the best transition, closest intensity match to speech, and lowest time alignment cost.

Recent work has employed motion graphs for gesture synthesis in the case of dyadic conversation behavior. For a dataset of 30 minutes of conversational data, Yang et al. [3] segment speech audio into phonemic clauses (groups of words with one strongly stresses word), as well as marking hesitation pauses and listener response. Motion data is annotated with the gesture stroke timings and used to build a motion graph. For gesture synthesis, a stochastic greedy search is then used to search the motion graph given a set of associated speech constraints. For synthesizing natural motion without jumps, a coherent sequence of poses must be created. To achieve this, the work finds a path through connected states in the motion graph, while matching the input speech constraints with the original associated audio of a state sequence. The search through the motion graph is illustrated in Figure 2.11.

Building a motion graph for very large datasets can potentially become problematic; Yang et al. [3] note that the motion variety of conversational gesture behavior requires a much larger graph than, for example, was previously used for locomotion, even for their relatively small amount of data of 30 minutes. Constructing and searching a motion graph for multiple hours of data would require questionable computing power. Some

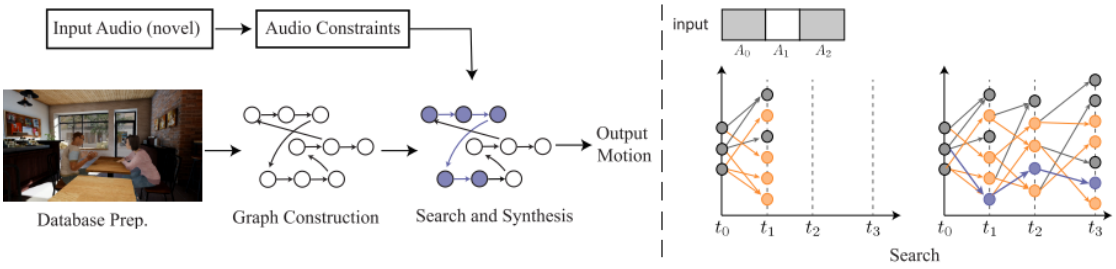


FIGURE 2.11: Gesture synthesis method by Yang et al. [3]. Left: overview of their method. Right: Visualization of a search through the motion graph (orange are considered low-cost action options and purple is the final selected path). (Images from Yang et al. [3])

alternative methods for connecting parts of natural motion into sequences were discussed in Section 2.1.3.

As the presented statistical models work with annotated datasets, a central limitation is the ease of extending these models; modelling any new speaker or gesture style requires many hours of hand labour.

2.8.3 Machine learning for gesture generation

Machine learning approaches for gestures can learn from motion data produced by conversing humans and, like statistical models, do not rely on the explicit formulation of if-then rules. Unlike statistical models, they often work without the need of any kind of hand-annotation. Instead, they implicitly learn the relationship between an input speech signal and the desired output motion. These approaches have largely focused on the generation of so-called beat gestures, simple, repetitive motions that mark speech rhythm. The kinematics of beat gestures (e.g. speed and acceleration) have been shown to correlate with the prosodic features of speech. Prosodic speech features can be extracted fast and easily, and contain information such as emphasis and emotion. Beat gesture systems rely purely on analysis of such prosodic features from speech without extraction of semantic content. The reason for the focus on beat gestures in this domain is based on the training mechanisms of machine-learning models, requiring relatively large amounts of examples for each "rule" to be learned. For example, to model the relationship of "you" and a pointing gesture, the training corpus should firstly contain numerous instances of "you" co-occurring with the desired pointing gesture. Secondly, the semantic tags, such as "you", have to be additionally extracted by creating a transcript of the input speech. Hence, data-driven approaches often neglect iconic, metaphoric, and deictic gestures. While this means that the generated gestures may not aid understanding, as semantically meaningful gestures might, they can make the agent appear more life-like and engaging.

Especially early works on gesture generation with machine learning have employed graphical models; in later works, neural networks have become widely popular. While neural networks can be seen as a special type of graphical models, we will not go into detail on mathematical definitions here.

2.8.3.1 Graphical models

Graphical models employ a directed or undirected graph structure to model a probability distribution, and hence represent a combination of graph theory and probability theory. Graphical models are trained by estimating likelihoods and inference is probabilistic. An advantage of graphical models is that they can model dependencies in the data even with small sample sizes.

Using a hidden Markov model (HMM), Levine et al. [111] presented a system relying solely on automatically extractable prosodic markers of speech. Based on the knowledge of gesture phase dynamics (refer to Section 2.5.2), the authors automatically segmented 30 minutes of motion capture data by tagging shifts between slow and fast motion. A clustering algorithm was used on the resulting segments to determine a number of recurring gesture sub-units (20 for the head, 45 for the arms, 6 for the lower body). For the concurrent speech, the pitch, intensity, and syllable duration were extracted. An HMM was trained to capture the relation of speech features and animation segments. (A similar approach was also later presented for Turkish conversational data [112].) The proposed system works in real-time by predicting syllable peaks. A new gesture was set to begin at the detected syllable peak, meaning the previous gesture ended at or before this syllable peak and honored the synchrony constraint. The authors reported problems with overfitting (lack of generalizability to new data) and proposed an alternative approach of representing gesture motion through kinematic features and combining an HMM (to model the transitions of motions) with a conditional random field (to model the relation of speech features to motions) [113]. Six kinematic parameters were extracted for each segmented (determined by velocity changes, as before): Spatial extent, duration, velocity, acceleration, curvature, and hand height of the gesture. These parameters could later also be used for affecting style changes of the motion. In both works, the produced animation was one selected from the segment library, rather than a truly "generated" motion.

Moving away from predefined gesture segments, Chiu and Marsella [114] used hierarchical factored restricted Boltzmann machines modelling temporal patterns of gesture motion to generate new motion frame-by-frame. Based on wrist height, motion capture data was segmented into gesture and non-gesture motion, resulting in a small selection of less than 40 seconds of training data. The resulting gesture motions were relatively

rudimentary beat gestures. Follow-up work improved results by encoding the gesture segments in a low-dimensional space via Gaussian process latent variable models [46]. This low-dimensional embedding of the gesture space addressed the problem of finding and generating transitions between gestures; determining the transition within the embedding space allowed to take into account similarities in gesture dynamics and posture, creating trajectories honoring the constraints of natural human motion. A conditional random field was trained to predict sequences of labels indicating whether or not a gesture was occurring based on the input audio features; during synthesis, appropriate motion trajectories are then sampled from the low-dimensional space.

Going beyond beat gestures, Chiu et al. [115] modelled more complex gestures by adapting conditional random fields to deep learning and combining prosodic, semantic, and syntactic speech features. Annotating a video corpus of over 9 hours of conversational data, the authors marked the occurrence of about 800 different words, giving each word a part-of-speech tag, as well as marking the occurrences of 14 gestural signs, such as "beats", "wipe", or pointing gestures. Though reporting improved results compared to previous work on gesture prediction, the applicability of the work was limited by the small set of predefined gestural signs and the requirement for tedious hand-annotation to obtain these.

2.8.3.2 Neural networks

With increase of computing power complex models trained on large datasets have gained favor over the simpler graphical models. While neural networks represent more of a black box learning, they can work on imprecise data and capture more complex patterns than older graphical models, allowing modelling of large datasets with implicit structure. Once a neural network is trained, inference is deterministic, rather than probabilistic.

Recent work has focused on neural networks, specifically recurrent neural networks (RNNs) for speech-to-gesture generation. RNNs use recurrent connections between network activations at consecutive time-steps to model data with temporal dependencies (see Section 1.2.1). Through this, RNNs can capture the dynamics of a motion pattern well, successfully modelling patterns of human motion modelling [116, 117]. RNNs have powerful modelling capacities, but due to a relatively large amount of model parameters,

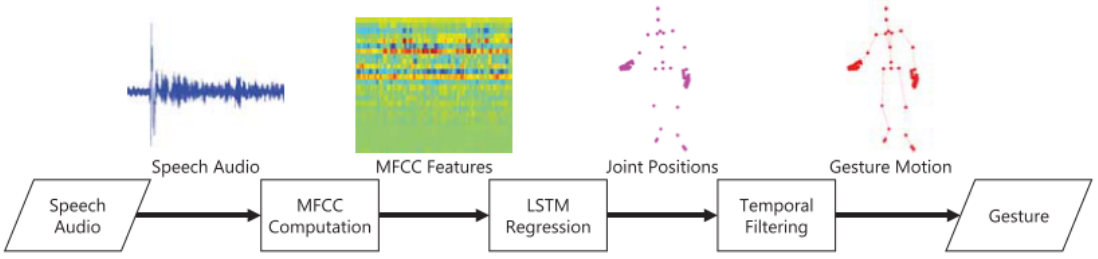


FIGURE 2.12: Machine-learning method for gesture generation by Hasegawa et al. [4].

they also require larger amounts of training data than previous proposed models (such as Gaussian process latent variable models).

Based on the DeepSpeech network architecture, successful for speech recognition, Takeuchi et al. [118] employed a recurrent network to model the speech-gesture relationship of a 2-hour Japanese conversational dataset containing largely metaphoric gestures. Input speech was automatically processed to extract the Mel-Frequency Cepstral Coefficients (MFCC), and the model outputs the rotations of 51 joints describing a human skeleton. The work was mostly unsuccessful, with the generated gestures receiving generally low scores of perceived naturalness, and timing and content appropriateness, outperformed on each score by mismatched real motion capture. In follow-up work, the authors proposed several changes, using joint positions and their velocity instead of rotations, as well as applying temporal smoothing to the output to address jittery and discontinuous motion [4] (see Figure 2.12). The generated gestures were able to outperform mismatched real motion on the previously mentioned three scores, though problematic phrasing of the naturalness questions were revealed. Specifically, users were asked to judge the smoothness of the motion, however, real motion capture data was somewhat jittery, whereas this was addressed in generated output through the temporal smoothing. The model was further developed in Kucherenko et al. [119], employing autoencoders to learn each a low-dimensional representation for the motion capture and speech data. Following the autoencoder training, the speech to gesture system was then trained to model the relationship between speech encoding and motion representation. The model showed improvement only regarding the naturalness dimension.

One problem in training RNNs is an observed tendency to converge to the mean pose in the data, leading to damped, constrained and somewhat lethargic motion close to the average pose. It was proposed that this may be due to error accumulation when feeding generated output back into the network [25]. Instead opting for a simpler feed-forward

architecture, Kucherenko et al. [120] presented a model combining both acoustic and semantic speech information. To capture the temporal dependencies between frames marking continuous motion (previously handled by the recurrent connections in an RNN), the authors used autoregression to feed generated frames back into the model input. Using the dataset presented in Section 3.1, speech data was transcribed and words were encoded with a pretrained language model, and prosodic information was represented by high-dimensional log-power mel-spectrogram features. The authors aimed to reduce motion-dimensionality by applying Principal Component Analysis (PCA), but found a negative effect on perceptual results. They further proposed to address issues with slow motion output by modelling velocity in addition to joint positions, however, this was unsuccessful. As seen in previous work employing RNNs, they reported their proposed model to move more slowly than the true motion, producing gestures closer to the mean pose. The authors did note a benefit of including semantic speech information, with the inclusion of semantics improving all four scores, human-likeness, semantic accuracy, helpfulness, and timing.

Aside from error accumulation through autoregression, a major problem in training neural networks for speech-gesture generation is the optimization method. The previous reported works here used a standard regression loss, essentially computing the numeric difference between a predicted and true pose or pose sequence. For speech gesture, however, as previously discussed, there is no one "correct" gesture for a given utterance, rather, it depends on the speaker and their state of mind. Therefore, even when a gesture model during training predicts a gesture that would be judged as valid by an observer, if this gesture is numerically far from the true training sample, the model will receive a large error feedback. Short of guessing exact correct joint configurations, the model can minimize its error by keeping prediction around the mean pose, minimizing the error across all possible true poses.

Generative adversarial networks (GANs) have been proposed as one alternative training paradigm. Here, instead of minimizing a standard error function such as the mean squared error (MSE) of joint positions or angles, the model's objective is to produce output that is qualitatively similar to real data, as judged by another model, the discriminator, that is trained simultaneously in conjunction with the generator (see Section 1.2.2). GANs have been successful in human motion modelling tasks [121, 122], as well as in a head motion from speech generation task [123]. Based on this, Ginosar et al.

[124] proposed a convolutional network combining a standard L1 regression loss with an adversarial discriminator for predicting 2D gesture motion from speech [124]. To address the need for temporal continuity of the motion output, the model took as input the entire speech segment for which to create gesture output, generating one continuous stack of output poses. This meant that during prediction the entire utterance had to be passed in at once to produce temporally consistent output. The authors represented audio visually as a spectrogram, which was encoded by an audio encoder and subsequently processed by a UNet translation architecture [125]. The authors created a large dataset of over 140 hours of 2D pose keypoints extracted from YouTube videos of 10 speakers. (This work and dataset was not yet available at the time of our work). The speakers were professional performers, such as John Oliver (*Last Week Tonight*) and Seth Meyers (*Late Night with Seth Meyers*), producing largely rehearsed speech and generally producing a relatively small set of clear gesture motions. Their speaker-specific models generated sequences rated equally good as mismatched real gesture samples, as measured by the rate it fooled human participants. The failure to surpass mismatched real motion is an indication of the difficulty of the speech-to-gesture task.

2.8.3.3 Other generative models

While the idea of GANs is promising, in practice, they have been shown to be unstable to train through their implicit error feedback mechanism. To address this, an invertible neural network type called normalizing flows (NFs) have gained traction. NFs make it possible the compute exact output densities in an efficient manner, providing a more stable way of training. By modelling the non-deterministic probability distribution of the output, NFs do not rely on standard regression loss. Based on an NF model specific for human motion [126], Alexanderson et al. [127] proposed a model for gesture generation from speech, using the dataset presented in Section 3.1. By learning the next-step distribution of possible next poses, compared to standard regression training, the gesture output was more varied and could be re-sampled repeatedly. Therefore, inference was probabilistic again (as in graphical models), rather than deterministic (as in standard neural networks). This work much improved on previous problematic mean-pose regressed gesture output, however, the authors noted problems with overfitting, with output gesture behavior appearing less diverse and vivid than the true motion.

Lowering model complexity improved gesture diversity, but showed significant amounts of unnatural motion.

2.9 Evaluation methods for gesture generation

An integral part of designing machine learning generative models of gesture motion is the evaluation of the output, necessary both during development as well as for final performance assessment. Training a model requires the use of numeric measures, whereas judging the quality of the output of a trained model requires a perceptual evaluation.

2.9.1 Numerical evaluation

During development, rapidly computable numeric measures of quality are necessary for model optimization: The model's parameters are iteratively changed based on the computed error until convergence. A common choice is computing the difference between prediction and ground truth poses, computed frame-by-frame for a sequence of poses, either over joint positions (e.g. [4, 120]) or angles (e.g. [118, 128, 129]). Such numerical measures have also been used for reporting final performance of a model (e.g. [4, 120, 130, 131]), however, this can be problematic. For example, one solution minimizing average joint error is producing just the mean pose, the best solution for a model unable to understand speech-gesture correlations. This clearly is not a desirable output easily determined by an onlooker. Indeed, Kucherenko et al. [132] have shown that numeric measures of performance do not relate to this onlooker's judgement. Final evaluation of a model should therefore include subjective judgements from a perceptual study.

2.9.2 Perceptual evaluation

There are currently few guidelines or agreed methods for evaluating conversational agents, and gesturing agents specifically, highlighted in a review by Wolfert et al. [133]. We discuss some current methods below.

One of the first choices to make is the character on which to display the motion to be assessed (see Figure 2.13). Some works have used simplistic stick figures [118, 119, 124]; while this may focus judgment on the displayed motion alone, it may be hard to

FIGURE 2.13: Different levels of model realism used in gesture generation evaluation studies. (1) Takeuchi et al. [118], (2) Kucherenko et al. [132], (3) Sadoughi and Busso [134], (4) Yang et al. [3], (5) Huang and Mutlu [138].

conclude how suitable the motion is for a production character. Other works have used increasingly realistic models, from 3D models of a lay figure [127, 132], to low- [134] and higher realism humanoid models [3], as well as robots, both as 3D models [135] as well as physical robots, both video-taped [129, 136] as well as "live" [137, 138].

Regarding the experiment design, a wide variety of setups has been used with very little coherence between researchers. Takeuchi et al. [118] used 7-point scales asking participants to rate 3 concepts, the naturalness, timing, and semantic appropriateness of the gesture motion; Hasegawa et al. [4] asked 9 questions belonging to these 3 concepts, again on a 7-point scale. Sadoughi and Busso [134] used 5-point scales for 2 questions, naturalness and appropriateness; Yang et al. [3] presented stimuli pairwise, asking participants which motion they preferred; Kucherenko et al. [132] used a MUSHRA test (MUltiple Stimuli with Hidden Reference and Anchor), presenting 6 stimuli at once and asking participants to rank each of them on a 100-point scale with 5 labelled intervals. This design variation between studies can make it difficult to compare results.

Finally, there is variation in choice of experimental conditions, i.e. what to compare the generated motion to. As an upper bound, the ground truth motion is a natural choice. Additionally, comparing to mismatched motion-capture is a common choice: Real motion displayed with speech from a different recording (used for example in [4, 118, 132]). Interestingly, while this baseline has no speech-gesture coherence (other than by chance), it has been shown to be hard to out-compete with machine-learned gestures. As lower baselines, some works have therefore used ablated versions of their models for comparison [3, 119, 120, 127] or models from previous works [132, 139]. Comparing results to previous works is useful for determining advancements in the field; however, challenges include the availability of code for a published model and data, as well as the potential need for tuning others' models to a new dataset (if used) by

adjusting hyperparameters for fair comparison. By making our speech-gesture datasets open source, we hope to improve reproducibility of future work, advance the field, and enable fairer evaluation in future studies.

Chapter 3

Data Collection

A key factor in any machine learning or data-driven, approach to gesture generation is acquiring an appropriate dataset to model. Previous research into speech gesture generation has used small datasets [46, 114], non-English datasets [118], hand-annotated video data [115, 140], or 2D rather than 3D motion [124, 141]. Advances in machine-learning can only be harnessed fully with significant amounts of training data. For developing a speech-to-gesture generation model, we required a dataset of synchronized speech recordings and motion-capture. No large, freely available corpus of such multi-modal recordings of English conversational speech was known to us and we therefore recorded our own datasets for the purpose of modelling the speech-motion relationship.

When creating a corpus of speech gesture, an important consideration is whether to focus on a single speaker or diversify with multiple speakers. As we have discussed in Section 2.6 in the previous Chapter, gesture expression may vary significantly between different speakers. A problem in modelling multiple speakers in a data-driven approach is that it may be difficult to implicitly capture the relation of speech to gesture (what we want to capture) when this relation may be additionally modified or varied by person-specific factors such as personality traits. Each person may have their own specific way of matching gestures to speech. Therefore, we consider the simpler problem of modelling a single person's gestures before tackling the problem of generalizability to other speakers or style-control. Furthermore, previous work has also shown that gestures generated by models based on a single speaker are preferred by subjects [107], additionally motivating the focus on single speakers.

TABLE 3.1: Details of the two recorded datasets.

	dataset 1	dataset 2
hours	4	6.2
sequences	23	25
sequence length (minutes)	10	10-20
frame rate (frames/second)	59.94	120
number frames	877.5K	2,66M
recordings	motion, audio, full-body video	motion, audio, full-body video, facial video
content	personal stories and interests	personal stories and interests, sports

A second key matter in designing a gesture corpus is determining the context of the gesture behavior to record. As discussed in Section 2.6.3, gesture behavior may vary depending on the emotional state of the performer. One could aim to model gestures for different emotional states by working on a dataset of diverse emotional expressions, capturing the ways gesture is expressed in a happy, sad, angry, or other state. However, most of our conversations are conducted in a relatively neutral emotional state. Furthermore animated agents are often designed for tasks in which a calm, non-emotional behavior is preferred, such as real estate showings [142], museum guiding [5], counselling [143], or teaching [102]. We therefore decide to focus our gesture corpus on everyday, calm, conversational mode, without extreme emotions. Recordings were unscripted; the actors produced spontaneous, unrehearsed speech and gesture motion, allowing us to capture natural conversational behavior.

We recorded two datatsets of one speaker each, detailed below, with an overview in Table 3.1.[1] The two speakers exhibit distinctly different gesture style. In future chapters, we will present comparisons of the gesture performance of these two speakers. In Chapter 5, we analyze the phase structure of the speakers' motion, and in Chapter 6, we compare differences in gesture stroke expression.

We released our datasets to the wider research community at

`https://trinityspeechgesture.scss.tcd.ie/`.

[1]Dataset 1 was published with the listed Publication 1. Dataset 2 is being published with the listed Under Review Publication 1

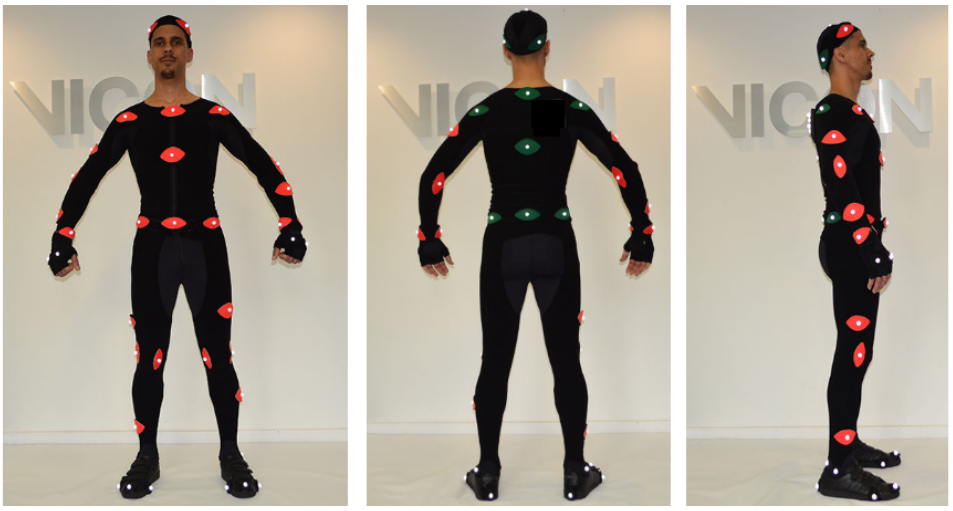


FIGURE 3.1: Marker setup for body motion capture. (Image by Vicon®)

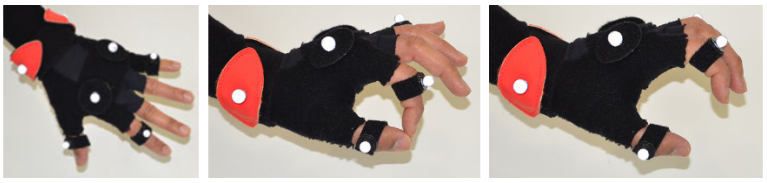


FIGURE 3.2: Marker setup for finger motion capture. (Image by Vicon®)

3.1 Dataset 1

We invited a single male actor for multiple recording sessions. The actor was an male native English speaker producing spontaneous and natural conversational speech without interruptions, i.e., without verbal cues from a conversation partner. The actor was instructed to speak freely and spontaneously about any topic he chose, including hobbies, daily activities, and movies. The actor speaks in a colloquial manner with a happy disposition, includes a large amount of gesture motions, and appears very animated overall.

The actor was addressing a person situated behind the camera in order to give him the visual feedback of a conversation partner. Each recording take was about ten minutes long. We captured 23 takes, totalling 244 minutes of data. (Two additional takes of eight minutes each are available of video and audio data, without motion capture).

The actor's motion was captured with a 59 marker setup and 20 Vicon cameras at 59.94 Frames per Second (FPS). The body marker placement is shown in Figure 3.1 and the finger markers are shown in Figure 3.2. Audio was recorded at 44 kHz. Video was

captured with a single HD camera placed between the actor and the person he was addressing (see Figure 3.3).

FIGURE 3.3: View of the camera in the dataset 1. The actor is always addressing a listener situated right behind the camera. The actor was allowed to move freely with the restriction to stay in good view of the camera.

3.2 Dataset 2

As described above, the speaker in dataset 1 moves in an overall very animated way, making it difficult to separate gesture motion from other body dynamics. For our second dataset, we decided to record a different style, with a much more grounded, calm, base motion, and defined, separable arm gesture movements. We chose an actor with naturally frequent gesturing behavior while exhibiting little lower body motion.

We again used a single male native English speaker for the complete recording. The speaker was unaware of the purpose of the recording and produced spontaneous, conversational speech without interruptions, i.e., without verbal cues from a conversation partner. The actor's instructions were the same described in Section 3.1, namely freely choosing topics to speak on and addressing the person behind the camera.

We recorded 25 takes, ranging between 10 and 20 minutes each, totalling over 370 minutes (more than 6 hours) of data. The actor's motion was captured with the same

59 marker setup and 20 Vicon cameras but at 120 fps (frames per second). Audio was recorded at 44 kHz. Video was captured with two cameras, one capturing a full body shot and the second camera capturing a higher-quality close-up shot of the face and parts of the upper body (see Figure 3.4).

FIGURE 3.4: Capture setup and video framing in dataset 2. We recorded both full body video (left) as well as close-up video (right).

Chapter 4

Motion and Speech Modelling

In this chapter, we explore the use of modelling the modalities of speech and motion before training a speech-to-motion task.[1] Drawing on previous research on motion and speech modelling, we employ transfer learning for a speech-to-motion model and assess the benefits.

4.1 Introduction

As shown by some of the previous research presented in Section 2.8, gesture generation has been proven to be a very complex problem. Direct end-to-end learning of speech to gesture motion has shown minimal success, resulting in overly smooth motion close to the mean pose. We therefore sought alternative approaches to this method. We hypothesized that results may be improved by modelling the gesture motion space first in isolation, before training the speech to gesture mapping. For this, in Section 4.3, we train a motion-to-motion task to yield a motion embedding to be reused for the speech-to-motion task. Similarly, in Section 4.4, we then investigated the benefit of prior speech modelling. Here, we integrate a pretrained model for speech recognition into the speech-to-gesture model. The model knowledge about the motion or speech distribution gained from pretraining is added to the speech-to-gesture model through transfer learning.

Transfer learning allows knowledge gained from training one problem to be applied to a related problem. For example, an image classifier trained on a large publicly available

[1] The contents of this chapter were published at the ACM International Conference on Intelligent Virtual Agents 2018 (IVA'18) (Listed Publication 1)

image dataset could be re-utilized for learning to label objects in a smaller, domain specific dataset. Transfer learning has shown success in a variety of domains, such as machine translation [144], image classification [145], and visual emotion recognition [146]. Transfer learning is actualized by using the model weights resulting from training a previous task as weight initialization for a new task.

We investigated the benefits of this knowledge transfer in the speech to motion task in two studies. First, in Section 4.3, we applied knowledge from human motion modelling, second, in Section 4.4, we employed language modelling.

4.2 Data processing

We used dataset 1 presented in the previous chapter in 3.1; dataset 2 was not yet available at the time of this research. We explored different representations of the motion data for our learning task, such as Euler angles and quaternions, but finally used the raw joint angles in exponential map format, as proposed by Fragkiadaki et al. [147] and used by Martinez et al. [148]. Two takes were selected as validation data, representing about 8% of the total data. During each validation step, 8 seeds are randomly selected from the validation set to compute the validation loss.

We considered different audio features to represent the speech signal, but in the final version of this work used the log of the 27 values of the Mel-scaled spectrogram with no cosine transforms, computed from FFT magnitude. The more primitive mel-frequency filter bank values have outperformed MFCC features in previous works in both pure speech modelling [149] and speech to motion modelling [150]. We extracted the audio features with openSMILE [151].

4.3 Motion model transfer learning

We drew upon the extensive research done in the area of human motion modelling for attempting improvements in speech-to-motion modelling. We hereby considered gesture generation from speech as a transfer learning task where a model is pretrained with a simpler motion-to-motion task before training the final speech-to-motion task. Our pretraining task was predicting the next frames of a motion sequence, based on the input

of a preceding motion sequence. This requires modelling of the dynamics of the motion modality which we hoped would give an advantage in tackling the later cross-modality task.

Current work had shown the potential of recurrent neural networks for modelling human motion [147, 148, 152]. Recurrent networks can model sequential data by using recurrent connections between network activations at consecutive time steps. For human motion modelling, recurrent networks seem to be able to capture the dynamics of a motion pattern well. Here, we applied the motion model proposed by Martinez et al. [148], which has yielded good results with a relatively simple and fast-training architecture. We hoped that applying previous knowledge about the motion domain could decrease the complexity of the notoriously hard to model speech-to-motion relationship.

4.3.1 Model architectures

We experimented with two variations of a sequence-to-sequence architecture for learning the speech to motion prediction task. In the first setup, we predicted a motion sequence directly from a speech sequence, with a direct transition between encoder and decoder (model 1 in Figure 4.1). In the second setup, we inserted an additional recurrent embedding layer between speech encoder and motion decoder (model 2 in Figure 4.1). Our architectures are based on the motion modelling network proposed by Martinez et al. [148]. We downsampled our data to 50 frames per second.

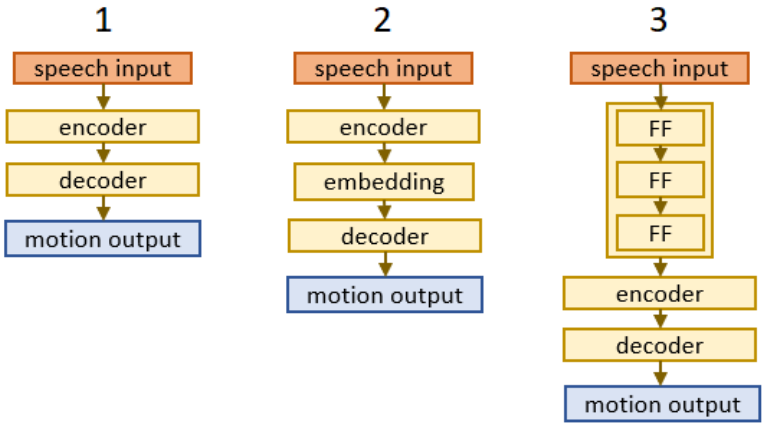


FIGURE 4.1: The three network architectures. For models 1 and 2, the decoder was pretrained with a motion modelling task. For model 3, the feed-forward (FF) block was taken from the pretrained DeepSpeech language model. All encoders, decoders, and embeddings are a GRU cell of size 1024.

4.3.1.1 Speech to motion

Both encoder and decoder consisted of a single recurrent cell, specifically a gated recurrent unit (GRU), of size 1024 with residual connections between the output and input of the cell. The residual connections are skip connections that allow a copy of the input to skip the encoder cell to directly feed into the output of the cell. The loss was computed as the mean squared error between the predicted motion sequence and the ground truth in angle space.

We first pretrained the network on a motion prediction task with residual connections modelling motion velocity, as described in Martinez et al. [148]. However, we did not tie weights between encoder and decoder, allowing encoder and decoder weights to be updated separately. We trained this sequence-to-sequence architecture to predict the next 15 motion frames based on an input motion sequence of 200 frames. We used our complete set of motion data for this training task. We trained the network for 100k iterations, results are visualized in Figure 4.3. We then took the learned weights of the decoder and reused them for the final speech-to-motion training task.

In the speech-to-motion training task, we predicted 20 motion frames based on both the corresponding 20 audio frames and 180 preceding audio frames (see example sequence in Figure 4.2). Hence at each prediction step, the network gets a context of 4 seconds, and predicts the final 0.4 seconds of gesture motion. We empirically found prediction of longer motion sequences hard to learn, but outputs of multiple overlapping speech sequences could theoretically be concatenated into longer motion sequences.

We ran the same network without motion pretraining to evaluate the benefits of transferring the motion model knowledge to the speech to motion models.

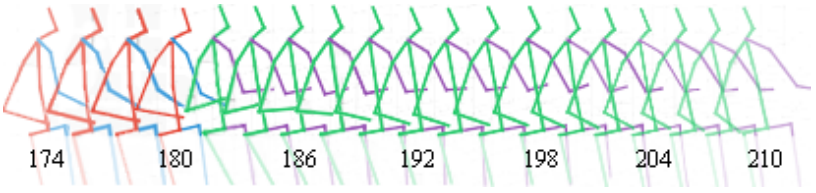


FIGURE 4.2: Example of a predicted motion sequence from the speech-to-motion model. Annotated is the respective frame, prediction starts at frame 180.

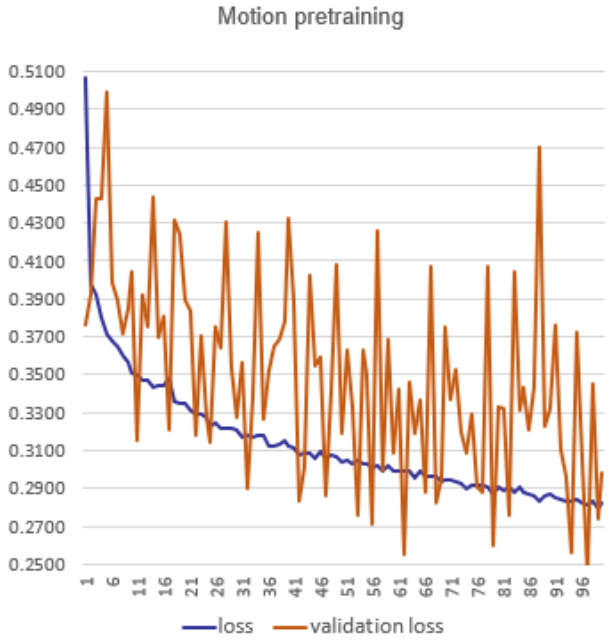


FIGURE 4.3: Results of motion pretraining. Plotted is the mean angle loss over one evaluation period, and the validation loss at the evaluation step. One evaluation period consists of $x * 1k$ iterations. We pretrain the model for 100k iterations.

4.3.1.2 Deep speech to motion

In our second architecture, we added an additional recurrent layer as connection between audio encoding and a motion decoding. The additional layer consisted of a single recurrent cell (also a GRU) of size 1024, also with residual connections. The output state of this additional cell was passed to a motion decoder that had been pretrained on a motion modelling task as described in the previous section. As previously, we compared the results of the pretrained versus not-pretrained model.

4.3.2 Results

We pretrained the network for an initial 10k iterations as suggested by the model's authors [148], we then continued training for 1k iterations at a time while checking for improvements. The results of this process are plotted in Figure 4.4. We trained the model for a total of 100k iterations, after which training slowed down and significant further improvement would take an large amounts of time. We used the resulting model weights for initializing the decoder of our speech to motion models.

We trained both of our final architectures for 100k iterations. For the deep speech to motion architecture, we first tested fixing the weights of the motion decoder while only training the preceding layers in order to focus on learning a motion representation from speech. However, this did not result in a good learning trajectory and we hence opened all model weights to updating during training. Though loss and validation loss as plotted in in Figure 4.4 still appear to be dropping for both models after 100k iterations, training has largely stagnated at this point. Both networks reached a minimum of about 0.38 in angle loss error at this point.

We compared our pretrained networks' performances to the same architectures without motion pretraining. Surprisingly, as we can see in Figure 4.5, pretraining only appears to help at the beginning of the training, essentially speeding up convergence, before stagnating at approximately the same error value.

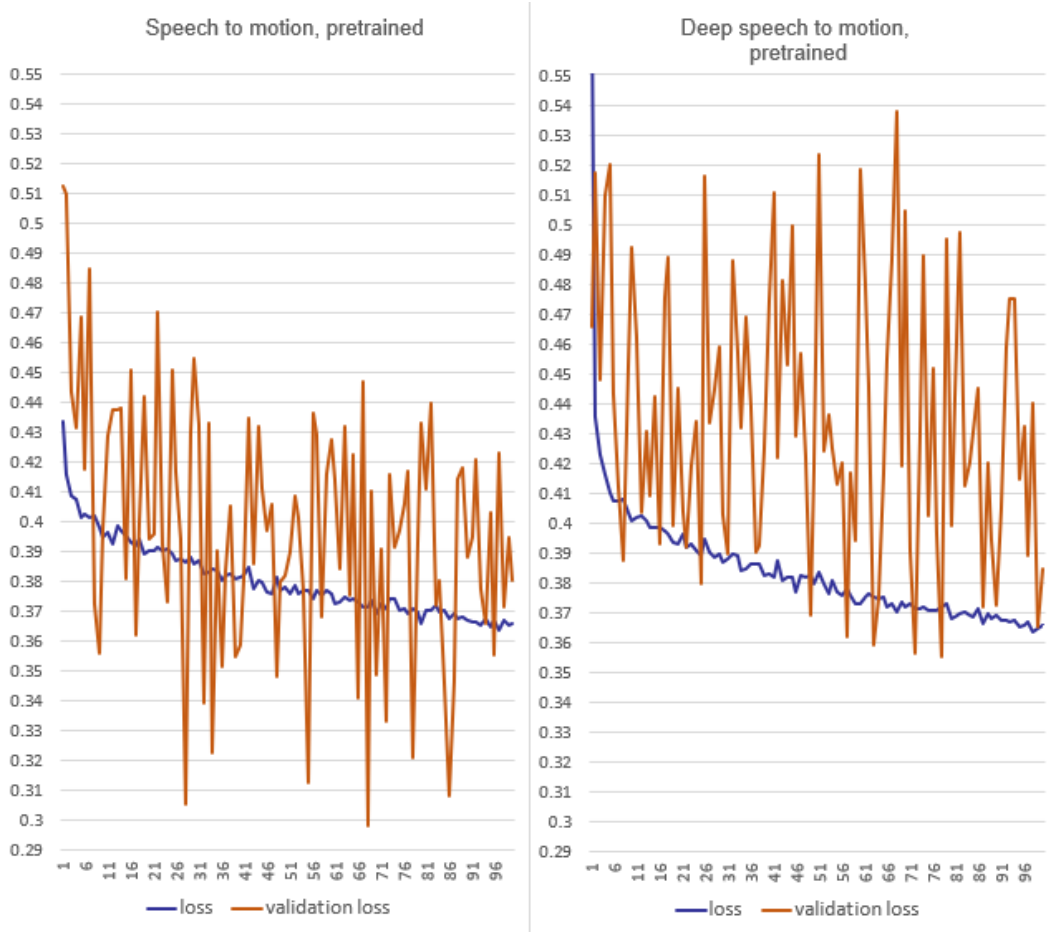


FIGURE 4.4: Training results of the speech to motion models after pretraining with motion modelling. Plotted is the mean angle loss over one evaluation period, and the validation loss at the evaluation step. One evaluation period consists of $x * 10^3$ iterations.

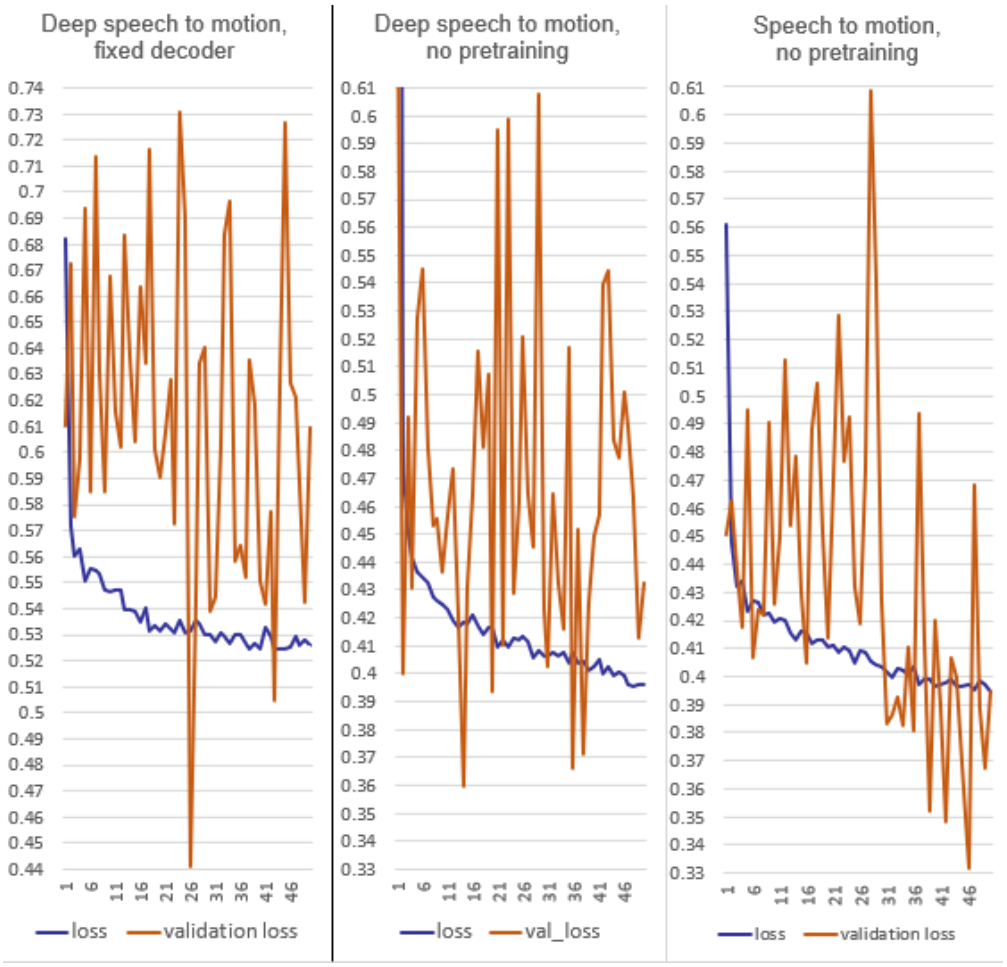


FIGURE 4.5: Pretrained vs. not pretrained results. *Left*: Results of training the deep speech to motion model with fixed decoder weights. *Middle and right*: Results of training the speech to motion models without prior motion modelling. Plotted is the mean angle loss over one evaluation period, and the validation loss at the evaluation step. One evaluation period consists of $x * 1k$ iterations.

4.3.3 Discussion

Our models reached a relatively low error for short-term motion prediction from speech, and we did not observe a significant difference in performance between our two network architectures. Both the speech to motion, and the deep speech to motion network converged to similar error results, though the validation loss in the speech to motion model appeared more promising. Notably, for the deep network, we were forced to open the decoder weights to further updating during training, as the pretrained weights did not yield good results. The poor performance of the pretrained motion decoder became further apparent when comparing network performance with and without prior motion modelling. We found no significant advantage of training our networks on a pure motion modelling task before training the speech to motion prediction task.

It is interesting that transfer learning of a motion model did not help our recurrent networks' performance. It is possible that the pretraining did not actually learn a good enough motion embedding. This would be supported by the finding that plugging the pretrained model's decoder into our deep model and only training the encoding and connection layer did not yield good results. Martinez et al. [148] do note in their work that aperiodic motion such as during a discussion (as opposed to periodic motion such as walking), remain hard to model in general. The dynamics of gesture motion might be better understood by focusing on the actual speech flow, as opposed to modelling an inherent motion representation. That is, gestural motion may not be well suited for modelling without the context of speech flow.

The work of Martinez et al. [148] also suggests that the motion model may be improved by jointly learning on multiple different actions. We only trained our model on gesture motion and hence incorporating data from multi-action databases could produce a richer embedding and improve the training outcome.

It is possible that the speech to motion learning task is not as related to the motion-to-motion task as we had assumed. The speech to motion 'translation' is arguably more complex than motion forecasting and might require much more than a rudimentary understanding of human motion dynamics.

We observed a major problem of mean pose regression: Model convergence during training inevitability came with a heavy collapse to the mean pose, resulting in few moving frames before the output regressed to the mean pose. This is to be expected somewhat when training the network on such a short-term prediction task, however, we also found longer sequenced empirically very difficult to train. While this is a common problem observed in motion generation with neural networks, it is also difficult to avoid. It has been attributed to error accumulation when feeding the network's output back into itself at test time [25], and newer work has addressed this issue with some modifications of the network training [153].

4.4 Language model transfer learning

We investigated the potential benefit of language modelling over motion modelling. This is motivated by viewing the speech to motion learning task as more of a translation

between two, if very different, languages, rather than a motion modelling problem. As the language model, we opted for a state-of-the-art speech recognition model that has been trained on a very large dataset, specifically, the mozilla's pretrained DeepSpeech model[2], which performs speech-to-text translation. We hypothesized that employing this language model could help reduce the complexity of modelling the speech to motion relationship.

4.4.1 Model architecture

In an effort to maximize comparability to the motion transfer learning approach, we kept as much of the network architecture and data processing as possible as described in the previous section. In the following, we describe the changes necessary for the language transfer learning.

The input data was processed for the DeepSpeech model, which takes at each time step the 26 Mel-frequency cepstral coefficients (MFCCs) of the current frame, plus 9 frames of context to each side, resulting in 494 input features.

As the DeepSpeech model was trained to map speech audio to text, we did not use the full model architecture, but rather extract the first few layers that map the audio features to an internal representation. Specifically, we used the first three, feed-forward layers and loaded their pretrained weights. We experimented with taking the outputs from other layers but the loss either did not converge or did not reach acceptably low values. The loaded weights were fixed in our task, i.e. they were not updated during training. Hence the training objective was the learning of a mapping from the speech representation of the DeepSpeech model to motion data.

The network architecture built upon the DeepSpeech processing was essentially the same as described in Section 4.3.1.1, consisting of an audio encoder and a motion decoder cell, each a GRU of size 1024 (model 3 in Figure 4.1). We again predicted 20 motion frames based on both the corresponding 20 audio frames and 180 preceding audio frames.

[2]https://github.com/mozilla/DeepSpeech

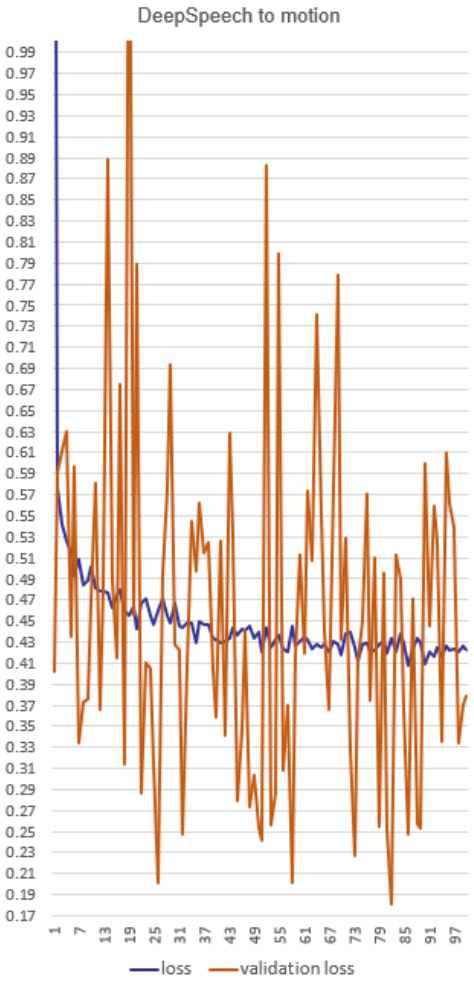


FIGURE 4.6: Results from applying the DeepSpeech model to our speech to motion task. The first three layers were pretrained with the DeepSpeech speech recognition task.

4.4.2 Results

In Figure 4.6, it is visible that the loss values flattened out relatively quickly during training, not reaching our previous results for models pretrained with a pure motion task (compare results to Figure 4.4).

4.4.3 Discussion

The language model pretraining did not yield improved results compared to the motion model pretraining. In fact, the error in angle space was higher still than the simpler model from the previous section without pretraining. This suggests that the speech representation learned from the speech recognition task of the DeepSpeech model is not

suitable for our speech-to-motion-task. One reason for this could be that the DeepSpeech model aims to recognize the phonemes (and from this the words) of an utterance, whereas the produced motion might rely more on prosodic information (emphasis, speed, ...) of the utterance, that is filtered out early in the DeepSpeech processing. Hence, if using a language model for transfer learning in our task, a better choice may be a model that has been trained on a task that relies on prosodic information rather than word recognition, such as emotion recognition. Is is however unclear how transferable knowledge from e.g. an emotion model is to neutral conversational speech.

4.5 Discussion

We did not observe a clear benefit of pretraining the speech-to-gesture networks, whether a motion or a speech modelling task was applied. We furthermore observed that during gesture generation, the output of all models quickly converged to a mean pose. This is a previously observed problem for recurrent networks trained with a standard error function. Longer term motion sequences have been shown to quickly regress to the average pose (such as in Martinez et al. [148] and Jain et al. [152]). This may be due to error accumulation when feeding generated output back into the network [25], resulting in damped motion that may look constrained and unrealistic.

Another potential cause of mean pose regression is the common training method of using a mean pose loss, such as the mean squared error used here. Due to the highly varied relationship of speech and gesture, it is difficult or impossible to predict the exact correct gesture for a speech segment. However, a standard loss function such as the mean squared error penalizes production of any pose sequence that is numerically far from the ground truth, even if an onlooker might judge the generated sequence as plausible for the speech. In the next Chapter, we will explore an alternative training paradigm to a standard regression loss, namely Generative Adversarial Networks (GANs). These aim to assess output in a more "human-like" manner by training a second network to make the judgment of plausibility, rather than providing an error representing the distance to the ground truth poses.

While the motion model itself might benefit from a larger variety of actions to model as previously mentioned, our speech to motion model might also improve with a larger

variety of conversational data. For this work, we only had data from one actor available and it is difficult to measure how well this specific person's gestures can be modelled. To reduce modelling complexity, we would like to isolate the actor's upper body joints to focus solely on his arm gesture motion. However, this is problematic for the actor used here, as he exhibits a large amount of whole body motion, so that isolating arm motion potentially removes important dynamic information. For this reason, we decided to look at a second actor with a more suitable gesture style in the next chapter.

Finally, the results for short term prediction indicate that gesture generation directly from speech does work to some extent; a basis upon which we built with major improvements in the following chapters.

Chapter 5

Adversarial Network Training

We have discussed that a key problem of modelling co-speech gesture behavior is its non-deterministic and variable nature that may not be captured with standard regression training. In this chapter, we explore the use of a generative adversarial training paradigm for mapping speech to gesture motion[1]. We defined the gesture generation problem as a series of smaller sub-problems, including plausible gesture dynamics, realistic joint configurations, and diverse and smooth motion. Each sub-problem was monitored by separate adversaries. For the problem of enforcing realistic gesture dynamics in our output, we trained classifiers with three different levels of detail to automatically detect gesture phases.[2] We hand-annotated and evaluated over 3.8 hours of gesture data for this purpose, including samples of a second speaker for comparing and validating our results. We find adversarial training to be superior to the use of a standard regression loss and discuss the benefit of each of our training objectives.

5.1 Introduction

The non-deterministic mapping of speech to motion means for one utterance, multiple variations of a gesture (or no gesture at all) may be perceived as plausible by an observer. This presents a difficulty in training a speech-to-gesture model; even a plausible produced gesture may be penalized when it is numerically far from the exact gesture found in the

[1]The contents of this chapter were published at ACM Motion, Interaction and Games 2019 (MIG'19) (Listed Publication 2), as well as in Computers & Graphics 89 (2020) (Listed Publication 3). The second author, Michael Neff, acted as an additional advisor while the implementation was conducted by myself.

[2]We release the classifiers for other researchers at `https://trinityspeechgesture.scss.tcd.ie/`.

dataset for this utterance. A standard regression loss in training a speech-to-gesture model is therefore not ideal.

In this chapter, we apply two novel techniques for training a recurrent neural network (RNN) producing gesture motion based on input speech. Firstly, we trained a speech-input-motion-output RNN in the manner of a generative adversarial network (GAN) instead of a standard regression loss, and we specifically used multiple adversaries instead of a single one.

Secondly, we studied the phase structure of a gesture dataset and trained a classifier to automatically detect these phases. The phase structure of natural gesture describes the dynamics and functions of motion segments within it, and can be divided into distinct parts: preparation, stroke, holds, and retraction (see Section 2.5). We aimed to capture these specific dynamic phases in our gesture generation system. While such phases are present in any natural gesture data, capturing the phase structure implicitly would arguably require a large dataset. Instead, we explicitly segmented the phase structure of gesture motion. The expression of these phases and their sequencing may vary from speaker to speaker, making their labelling a difficult and at times ambiguous task.

In an adversarial training paradigm, we used the automatic phase labelling to extract the phase structure of real and generated motion. Producing realistic phase structures becomes a training objective of the generator, enforced by a discriminator specifically designed for distinguishing phase sequences. The set of training objectives further included humanoid skeleton constraints, and utterance match and diversification objectives, each represented by separate discriminators. Our multi-discriminator design allows the gesture generation problem to be defined with multiple smaller sub-problems. We discuss how each of our discriminator objectives improves the final result.

We will first introduce the phase classifier in Section 5.3, before discussing the speech-to-gesture model in Section 5.4 and its adversarial training in Section 5.6.

5.2 Data processing

We wanted to isolate upper body motion and therefore restricted modelling to dataset 2 (see Chapter 3.2) in this chapter; dataset 1 proved very difficult to annotate due to

the actor's less distinct gesture motion and his frequent walking around and turning. However, we used samples of dataset 1 in Section 5.3 to validate our results.

We processed the recorded speech with openSMILE [151] to extract 26 Mel Frequency Cepstral Coefficients (MFCCs), as well as the F0 (pitch) value. MFCCs are commonly used in speech recognition tasks and the F0 value as a prosodic feature carries information about emphasis. Speech features were extracted with a window size of 20 ms at steps of 10 ms, resulting in data of 100 fps.

We down-sampled the motion capture data from 120 to 100 fps to match the speech features. We centered and locked the root node of the motion data to the origin position with zero rotation and then extract the absolute positional values of the captured joints. Our actor remains fairly static in his lower body and we were therefore able to capture most of his dynamics from the joints upward of the locked root.

We normalized all speech and joint position features to zero mean and unit variance. We trained all models on 20 fps; in order not to lose data, we took 20 fps data from 5 subsequent starting positions, resulting in 5 sets of 20 fps data.

5.2.1 Gesture phase annotation

We annotated the phase structure of a subset of 226 minutes of the complete dataset using the ANVIL annotation tool [154]. The 226 minutes were selected at random from the dataset. We aimed to annotate as much of our dataset as possible while ensuring annotation quality. For this purpose, we trained six annotators whose work was then repeatedly cross-checked at the start, before each annotator was assigned separate data clips. We annotated nine different gesture phases; (1) preparation, (2) stroke, (3) pre-hold, (4) hold, (5) independent hold, (6) rest hold, (7) partial retract, (8) retract, and (9) 'none'. Table 5.1 shows the frequency of each phase within the annotated data subset. *Pre-hold* and *hold* occur before and after the gesture, respectively. *Independent hold* occurs when a gesture has no stroke, but is defined by a held pose. *Rest hold* occurs when the hands are held in a relaxed position after a partial retract, without being fully retracted to the sides of the body. *None* occurs when no gesture is being performed; the arms are either fully retracted to the sides of the body or a no-gesture movement such

TABLE 5.1: Frequency of the 9 annotated gesture phases in the total annotation set of 226 minutes.

Gesture phase	Number of occurrences		Percent of annotated time	
	speaker 2	speaker 1	speaker 2	speaker 1
Preparation	5775	130	19.1%	14.9%
Pre-hold	979	17	3.2%	1.6%
Stroke	8655	160	39.6%	28.5%
Hold	5100	110	24.8%	26.1%
Independent hold	94	3	0.8%	0.7 %
Rest hold	474	27	3.1%	10.3%
Partial retract	1077	48	3.8%	6.5%
Retract	409	13	1.3%	2.1%
'None'	475	14	4.2%	9.3%
Total	23038	522	100%	100%

as a self-adaptor is occurring. An example of an annotated gesture sequence is given in Figure 5.1.

The speaker performs on average 38.1 gesture strokes per minute, or one gesture every 1.6 seconds. Assuming roughly the same gesture frequency in the remaining un-annotated 140 minutes of data, we estimate that our dataset contains approximately 14,000 gestures.

We computed pairwise coder agreement with ANVIL [154] by double-annotating five samples totalling 50 minutes of data, each with a different annotator combination. We found high segmentation agreement, averaging 98.5% (min=95.5%, max=99.9%), indicating high consistency in detecting phase boundaries. For the overall coding agreement that includes segment (or phase) labels, we achieved moderate agreement as defined by Krippendorff's alpha value [155], with a mean of $\bar{\alpha} = 0.46$ ($\alpha_{min} = 0.39, \alpha_{max} = 0.5$). As we pooled all hold categories for the phase classifier in Section 5.3, we compared Krippendorff's alpha value for the case of treating post-stroke holds, pre-holds, rest-holds and independent holds all as a uniform hold category: $\bar{\alpha} = 0.47, \alpha_{min} = 0.43, \alpha_{max} = 0.53$.

In order to evaluate the robustness of our automatic phase classification in Section 5.3, we annotated a short sample of gesturing of a second speaker. For this, we took samples of just under 5 minutes of data from dataset 1 (Section 3.1). This sample was not included in the training set and only used for evaluation. The speaker in this dataset exhibits a qualitatively very different gesturing style to the speaker in dataset 2. Speaker

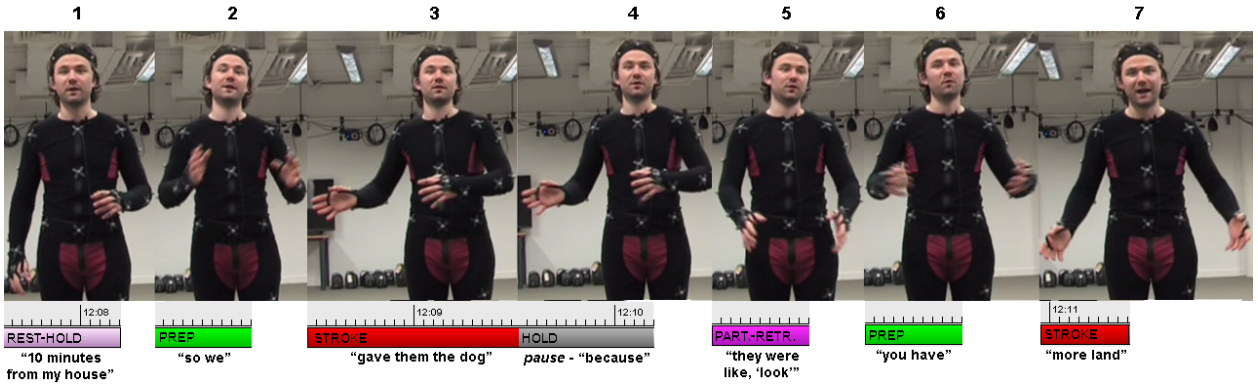


FIGURE 5.1: Sample of an annotated gesture sequence. For each annotated gesture phase, the speaker's accompanying phrase is given. (1) The hands start in a resting position. (2) The preparation phase brings the hands into position for the gesture. (3) The stroke phase carries the meaning of the gesture (the act of giving). (4) The hands stay in position, the speaker pauses for a moment. (5) The hands are retracted partially towards a restful position. (6) A new preparation phase immediately initializes the next gesture. (7) Another gesture stroke is performed, describing "more".

1 often incorporates the whole body in a gesture and rarely stands still. This means that extracting the motion of the upper body joints does not fully describe the performed gesture, some information will be lost. Hold phases mark another observable difference between the two speakers; whereas holds tend to be associated with minimal movement in speaker 2, the hold of speaker 1 appear overall less still, with the speaker in seemingly constant motion.

Our annotated sample of dataset 1 suggests a similar gesture stroke frequency as observed in our large-scale annotation of dataset 2; we calculated 33.6 gesture strokes per minute. We annotated 160 strokes in this sample. All annotated phase frequencies are reported and compared to speaker 2 in Table 5.1.

5.3 Phase classifier

Modelling gesture motion from speech directly is a hard problem. As described in Section 5.1, the same phrase may be plausibly accompanied by many different gesture shapes. Speech features may be more easily associated with the dynamics of gesture motion; the kinematics of gestures (such as speed and acceleration) have been shown to correlate with the prosodic features of speech [44]. However, implicitly inferring gesture dynamics from raw positional data may be difficult and require a large amount of data. We therefore modelled these dynamics explicitly. Namely, we extracted gesture phases

as higher-level representation of the characteristic dynamics of gesture motion. This representation is sufficiently low-dimensional (small set of different labels) to model its structure from a relatively small dataset. We hand-annotated the phase structure of 3.75 hours of data (as described in Section 5.2.1) and trained a classifier to detect gesture phases of a motion sequence. Our objective was to use this phase classification to enforce a realistic phase structure in the gesture generator's output. A classifier was necessary so that any new (un-annotated) motion can be segmented into phases and judged for its structural realism.

The phase structure variability within and between speakers previously discussed in Section 2.5.2, makes the task of automatic classification challenging, and, for a new, unseen speaker, particularly error-prone. Nevertheless, we consider even imperfect phase labelling a useful and reasonable way to explicitly describe different motion profiles present within a gesture, separating effortful, accented gesture strokes from less accentuated preparation and retraction as well as still hold phases. In this work, we focused on modelling just one speaker and his gesture dynamics to maximise training consistency of gesture dynamics in the training set.

The selection of gesture phases considered for enforcing realistic gesture dynamics is detailed below in Section 5.3.1, followed by a description of the classifier training (Section 5.3.2) and network architecture (Section 5.3.3). We include a robust 1-phase classifier for gesture stroke detection as a useful tool for future gesture analysis. The stroke phase represents the core, meaning-carrying part of a gesture, and hence its segmentation is essential for gesture form analysis. We validated all phase classification models on a second speaker with different gesture style (speaker dataset 1).

An overview of the phase classifier's role in the final architecture is shown in Figure 5.2, and will be discussed in more detail in Section 5.6.1.

5.3.1 Phase class simplification

The classifier assigned one phase label to each time-step of an input sequence. For training the classifier, we reduced the annotated gesture phase label set from nine to six classes that capture the main phase types by combining all types of holds into one class. This reduces the problem of unbalanced class frequencies (e.g. only 94 independent

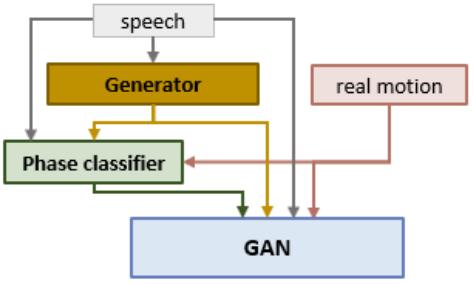


FIGURE 5.2: Overview of the system architecture. The generator receives speech features and produces gesture motion. The multi-discriminator GAN receives three different types of input: (1) the speech features belonging to a motion segment, (2) a motion segment (real or generated), and (3) the phase structure of the motion segment (determined by the phase classifier).

holds out of 23,038 phases), as well as removing some redundant information (e.g. a hold occurring between preparation and stroke can be assumed to be a pre-hold; a hold after a partial-retract is a rest-hold). Hence, we combined the labels 'pre-hold', 'hold', 'independent hold' and 'rest hold' into a super-class 'hold'. In effect, this simplifies the classification task by labelling all still frame sequences (sections with close-to-zero joint velocities) as one class, with the exception of the completely retracted 'none' position where the arms are relaxed by the side of the body. As discussed later, the partial-retract phase proved difficult to classify, so for training our generative network, we decided to combine it with the retract class, and due to its rarity we furthermore combine the fully retracted 'none' class into the retract group.

For our adversarial training we therefore had four phase classes: Preparation, holds (including pre-holds, independent holds, and rest-holds), strokes, and 'other'. The 'other' class combines retracts, partial retracts, and 'none' annotations. We believe this subset captures the most essential dynamics of gesture motion; we consider holds and strokes the most important representatives of gesture dynamics and their separation tends to get lost in standard training of recurrent networks (mean pose convergence leading to smoothed, damped motion). Second, we separated the preparation phase due to its high frequency and relevance in the gesture structure. Retracts are relatively infrequent for the speaker, as is the 'none' phase (completely retracted position); we decided to pool these classes together to make for a higher confidence model and a more achievable task for the gesture generator. The phase labels produced by the classifier are used as pseudo ground-truth during adversarial training, and we therefore need the classifier to be as confident as possible in its decisions.

5.3.2 Classifier training

Of our total of 226 minutes of annotated data, we separated 6.5 minutes of validation data by randomly selecting 13 start indices from which to take 30 seconds of data without overlap. Composing the validation data of snippets from multiple takes this way ensured that the validation performance is not annotator- or take-specific. The remaining annotations served as training data.

We trained three classification models for segmenting gesture. Firstly, we trained a 6-class model distinguishing all annotated phases (but pooling all hold categories), second, a 4-class classifier pooling rare phases into an 'other' class, and third, a 1-class model for detecting only the core stroke phase with increased confidence. For the stroke classifier, we predicted a single class, the stroke phase, which is the essential phase in gesture. This allows for more confident classification when dealing with different speaker styles, extending the applicability of this work. For all models, we trained a version each with and without speech pitch input. The pitch value captures information about speech emphasis and using a single speech feature ensures we are not increasing the input space significantly and hence minimize the network's ability to overfit. Including pitch improved our classification scores for the multi-phase models slightly (see Table 5.2), in line with the finding that speech is associated with gesture phase [156]. However, there was no apparent benefit of including pitch for the stroke classifier.

5.3.3 Classifier architecture

The classifier processed sequences of 100 time steps (5 seconds at 20 fps), and assigned a phase label to each step. The input of the classifier were the x, y and z directional velocities of 16 joints (total of 48 values, each normalized to a range of 0-1), corresponding to the shoulder, elbow, wrist, and each fingertip, as well as the corresponding pitch value. The directional velocity of a coordinate is:

$$v(x) = x_t - x_{t-1} \tag{5.1}$$

The two multi-phase models are visualized in more detail in Figure 5.3, but generally consisted of a two-layer recurrent network with an additional densely connected NN

(neural network) layer for input processing. The recurrent layers were Long Short Term Memory (LSTM) cells; specifically, a unidirectional LSTM in the first recurrent layer, and a bidirectional LSTM in the second recurrent layer. LSTM cells can handle sequential data, such as time series data, and bidirectional LSTMs specifically take both past and future data into account for predicting a time step. We regularized the network by applying dropout after each layer and batch normalization before the final output. Dropout rates were empirically determined to provide good performance without overfitting. The output layers of the multi-phase network applied a softmax activation.

The one-class stroke classifier is visualized in Figure 5.4. The architecture was similar to the multi-phase classifiers but used only a single recurrent layer, a bidirectional LSTM cell. The output layer applied a sigmoid activation in the single-class stroke classifier.

The differences in network architecture between the 3 classifiers resulted from empirically

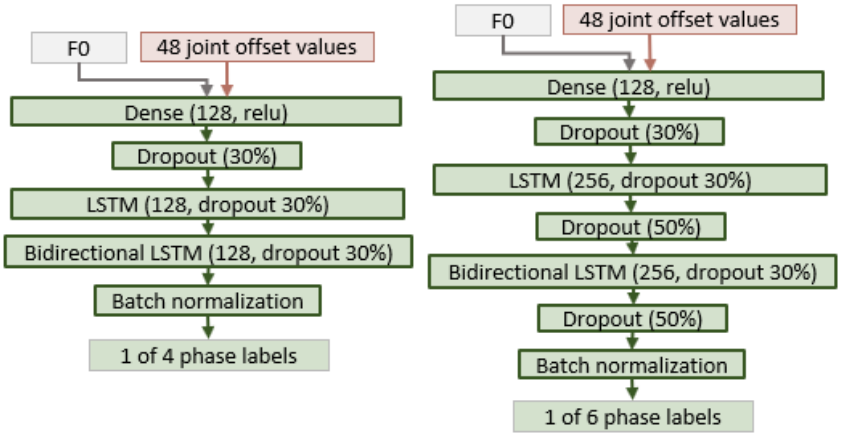


FIGURE 5.3: The two detailed network configurations for our 4-phase classifier and our 6-phase classifier. 'Dense' denotes a standard densely connected NN layer. In brackets are denoted the layer size or the dropout ratio. The 48 joint values refer to the x, y, and z offsets of the 16 joints shown in Figure 5.5.

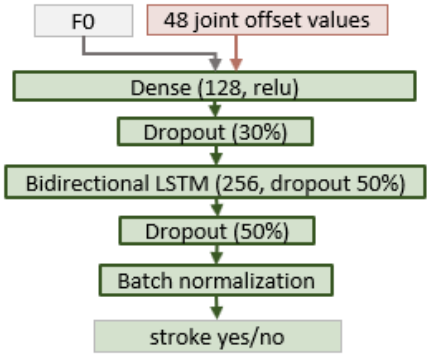


FIGURE 5.4: The network configurations for the 1-phase (stroke) classifier.

finding the best performing configuration for each number of classes. The number and size of recurrent layers was chosen based on the best found trade-off between modelling capacity and generalizability, i.e. reaching good performance without overfitting.

5.3.4 Evaluation

5.3.4.1 Multi-phase classifiers

The multi-phase classifiers reached an overall weighted F-score of 0.76 for both the 4-class and the 6-class model. The detailed results can be seen in Table 5.2. The stroke and hold phases reached the highest scores; this is likely due to both their distinct dynamics as well as their high frequency in the training set (see Table 5.1). Lower frequency phases with less distinct dynamics, such as partial retracts, were more difficult to detect. Furthermore, partial-retracts and preparation phases both averaged a length of less than 500 ms, making them potentially harder to catch as well as align; at our training sample rate of 20 fps, a prediction with just one frame of erroneous shift would only yield an 80% score. Notably, the annotated phase labels are only pseudo ground truth, as determined by an annotator, resulting in some inconsistencies and errors. Inter-rater category agreement for our evaluation samples averaged 64.4%, capping the realistically achievable score for the phase classifier.

Since the input was always a sequence of 5 seconds from a randomly drawn starting point, the classifier had limited context information for predicting the phase label of a time step. Providing the label of the phase preceding a sequence or increasing sequence length may improve classification results.

Validating our classifiers on the annotated sample of dataset 1 (denoted as 'speaker 1'), the 4-class model proved more robust with an F-score of 0.69. The 6-class model reached a score of 0.65, with the weakness lying in the less common classes, particularly partial-retract. The most confidently predicted class throughout all model versions and across both speakers was the 'hold' class; this may be the easiest class to extract as it contains almost all sections of zero velocity. Possible exceptions are the no-gesture sections annotated as 'none', though the speaker tends to swing his arms during these and indeed not stay still.

We compared results for the classification models, with and without speech pitch input (Table 5.2). For the multi-phase models (Table 5.2), the benefit of including pitch in the input to the classifier was more pronounced for the 6-class model, where all individual scores except 'partial retract' were improved by including pitch, as well as showing an improvement of 0.03 in the overall weighted F-score. For the 4-class model, the individual class scores improved (all except stroke) or remained the same (stroke), but the weighted overall F remained the same when including pitch as input. We also report the performance of the no-pitch models on the second speaker. No benefit is apparent for including pitch of the second speaker; this may be due to the articulation differences between the training and the validation speaker and using the pitch derivative instead could address this. For the stroke classifier, no benefit of including pitch is apparent for either speaker (Table 5.3); for the the validation speaker, using no pitch input yielded better performance.

We compared our results with the work of Madeo et al. [53] (Table 5.2), who employ a hierarchical strategy of single-class classifiers, where e.g. a hold classifier first detects all holds, subsequently a stroke classifier detects all strokes in the remaining data, etc. Their results represent the best scores across multiple models rather than a single model encompassing all gesture classes. That is, they trained combinations of single-class classifiers and the here reported results represent the highest scores for each class across combinations. For example, the model achieving the score of 0.79 for detecting a preparation phase is not the same model that achieves the score of 0.79 for stroke detection. Another significant difference to our work lies in the dataset composition; Madeo et al. [53] restricted the two captured participants to describing one of three comic strips. Their results indicated high dependence of performance on the comic story the classifiers were trained on (significantly reduced performance when training and test data were taken from different comic strip retelllings), as well as on which participant a classifier was trained on. As our dataset was captured across multiple days, with a large variety of spontaneous, uncued gestures, the performance of the classifiers presented in Madeo et al. [53] would likely not be adequate for this work.

TABLE 5.2: F-scores of phase classifier. Results without pitch input are reported in brackets behind the results with pitch input. Our 'other' class combined the labels *retract*, *partial retract*, and *none*.

Gesture phase	4 classes	4 classes speaker 1	6 classes	6 classes speaker 1	F-score Madeo et al. [53]
Preparation	0.64 (0.63)	0.56 (0.55)	0.65 (0.64)	0.56 (0.51)	0.79
Stroke	0.79 (0.79)	0.72 (0.7)	0.79 (0.78)	0.71 (0.71)	0.79
Hold	0.83 (0.82)	0.76 (0.76)	0.81 (0.78)	0.74 (0.77)	0.58
Partial retract	-	-	0.47 (0.49)	0.39 (0.35)	-
Retract	-	-	0.73 (0.70)	0.54 (0.52)	0.5
'None'	-	-	0.75 (0.56)	0.51 (0.59)	-
'Other'	0.64 (0.6)	0.58 (0.54)	-	-	-
Overall	**0.76 (0.76)**	**0.69 (0.67)**	**0.76 (0.73)**	**0.65 (0.66)**	

TABLE 5.3: F-scores of the stroke classifier.

Gesture phase	trained speaker	speaker 1
Stroke	0.79 (0.78)	0.72 (0.74)
No stroke	0.85 (0.86)	0.86 (0.88)
Overall	**0.83 (0.83)**	**0.82 (0.84)**

5.3.4.2 Stroke classifier

The stroke classifier reached a weighted average F-score of 0.83 on the speaker it was trained on (speaker of dataset 2), and a score of 0.84 on the validation speaker (speaker of dataset 1). Inter-coder category agreement for the case of stroke/ no stroke was naturally higher than for the full set of gesture phases, averaging 74.3%. Interestingly, it can be seen that the stroke classification score (first line in Table 5.3) was the same as in the multi-phase models for the training set speaker, reaching 0.79 for the training set speaker, and similar for the validation speaker, suggesting that we may have reaching the maximum score possible with an imperfect training set. The higher phase label consistency of the stroke training set may therefore be the main reason for the more robust classification.

5.3.5 Discussion

Looking at the relationship between the achieved F-scores and the inter-rater category agreement, we hypothesize that improving coder agreement would much improve classification results. We believe future improvements on the phase classification should focus on improving the training data consistency rather than the classification model.

The robust classification score of the stroke classifier for both the training speaker as well as the validation speaker makes it a good tool for future gesture analysis. As the stroke phase represents the essential, meaning-carrying part of a gesture, stroke segmentation is useful for additional information extraction such as gesture form detection.

It is less straightforward to train a classifier for other single phase types, as was done with the stroke present/ not present classifier. Since other phases occur less often across the training set, splitting our dataset into e.g. preparation/ no preparation would result in about a 1:5 ratio. Such unbalanced classifiers are more difficult to train, requiring a weighted loss function or an adapted (balanced) dataset (the latter resulting in a smaller training dataset).

'Hold' predictions may be more easily segmented by simply computing sections of close to zero velocity, and this could aid additional segmentation by an annotator as well as increase inter-coder consistency.

5.4 Gesture generator

The gesture generator was the core of our system and models the speech-to-gesture translation. The generator took speech features as input and produces the positions of the 21 joints shown in Figure 5.5.

5.4.1 Generator architecture

The generator received 27 speech features as input, composed of 26 MFCC values and the speech pitch (F0) value. The generator then inferred the x, y, and z positions of 21 joints: the hand, arm, and spine joints depicted in Figure 5.5.

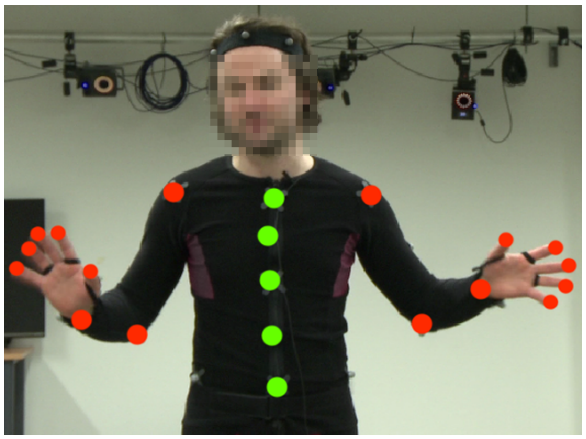


FIGURE 5.5: The 21 joints predicted by the generator.

The generator architecture is visualized in Figure 5.6. The speech input was processed by a densely connected NN layer (size 256, relu activation), followed by a dropout layer (30% during pre-training, 20% during adversarial training) and batch normalization. The network core was a Gated Recurrent Unit (GRU, size 256, dropout of 50% during pre-training and 20% during adversarial training). A GRU is a variant of a recurrent network cell with fewer parameters than an LSTM, allowing faster training. The output layer (densely connected NN layer with linear activation) of the generator produced the x, y and z position of 21 joints.

During pre-training (described in the below Section 5.4.2), the dropout rate was larger due to the MSE function used in pre-training posing a high probability of overfitting. The MSE gave the generator direct feedback on how far each predicted pose is from the ground truth. During later multi-adversarial training, the generator received less direct output feedback and was therefore less likely to be able to overfit on the dataset. The adversarial loss merely tells the generator the likelihood of the discriminator(s) finding its output to be real data, without per-pose numerical error feedback.

5.4.2 Generator pre-training

During later adversarial training (Section 5.6.1), the generator received feedback based on the phase structure of its motion output. This phase structure was determined by the phase classifier previously described in Section 5.3. The automatic phase classification meant that no matter what input, a phase label will be assigned to each time-step. Data points diverging from a skeleton structure and not resembling human motion could

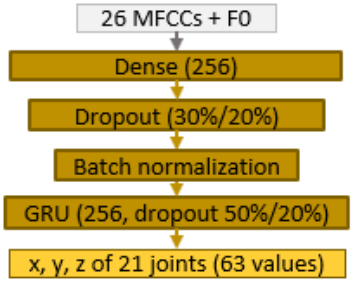


FIGURE 5.6: The generator network. The generator received 27 prosodic speech features (26 MFCCs + F0) and produced the xyz position of 21 joints. In brackets are denoted the layer size or the dropout ratio; the larger dropout ratios applied to pre-training with MSE.

get assigned an indeterminable phase label. We did not want very unrealistic data to be assigned a potentially realistic phase labelling. This could allow for the following scenario: the generator generates effectively noise, the classifier produces a realistic phase structure based on this, the generator receives positive feedback for having produced motion with a realistic phase structure. We therefore first ensured a quality baseline of generator output that can reasonably be assigned phase labels by the phase classifier. Hence, before adversarial training, we initialized the generator to a baseline output resembling a skeleton structure. For this, we pre-trained the generator with a standard mean squared error (MSE) loss of generated versus real motion:

$$MSE(m_g, m_r) = \frac{1}{T}\sum_{t=1}^{T}(m_g - m_r)^2 \tag{5.2}$$

MSE training allowed for fast convergence towards a skeleton structure, but as expected, this training suffered from mean pose convergence and produced only very damped motions around the average joint positions. This is visualized in Figure 5.7f, as well as in the supplemental video. We used this model as the starting point for the adversarial training, and utilized the training history for pre-training the phase discriminator as described in Section 5.5.1.

5.5 Adversaries

A training objective with a standard regression loss can be problematic for gesture generation due to the variability of speech gesture. The same or a similar utterance may reasonably be associated with various different gestures; the generator may produce a

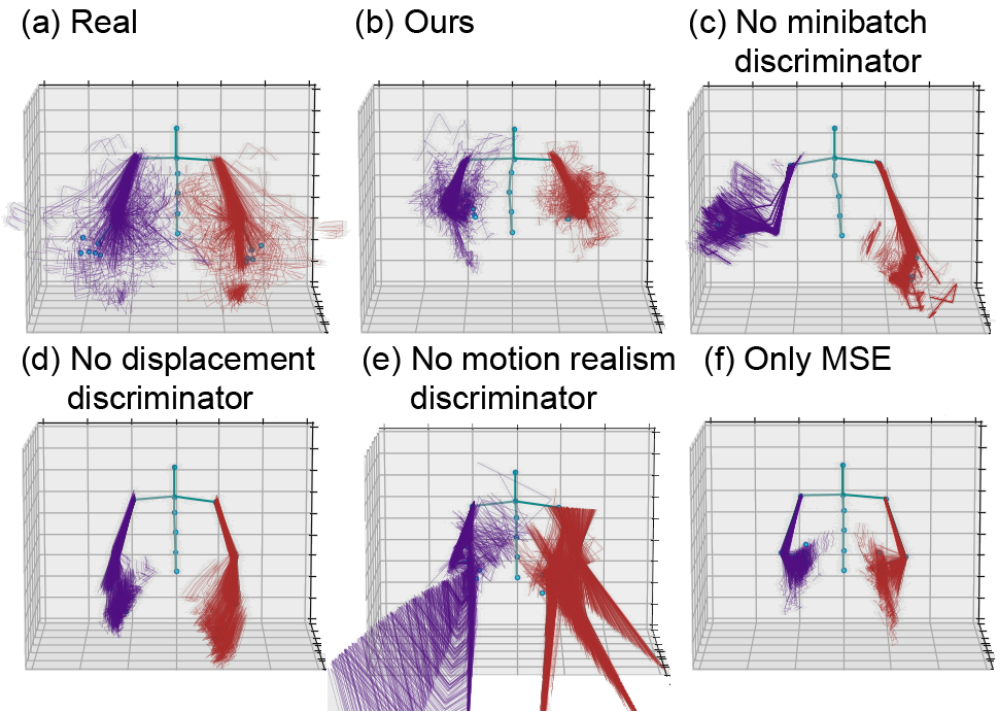


FIGURE 5.7: Motion distribution over 2 minutes, plotted at 4 fps. For one example clip, a) shows the real data distribution, b) the distribution with our method, c-e) examples of excluding specific training objectives, and f) the distribution for a model trained with a standard regression loss.

subjectively valid gesture that is nonetheless objectively far from the ground-truth pose sequence, resulting in a high training error. A common result is mean pose convergence, where the generator produces damped motion around the mean, minimizing error across all possibilities. Our adversarial training paradigm removed the tight constraint of predicting exact poses while still enforcing higher-level descriptors of natural gesture, as well as lower-level humanoid skeleton configuration constraints.

Specifically, in an adversarial training paradigm, the generator receives as feedback only a single value per generated gesture sequence, representing the decision of the discriminator whether the presented sequence looks real or not. Therefore, rather than receiving a numerical error for every pose in a sequence as is the case in a standard regression loss, the generator receives a single, more qualitative judgement about the entire pose sequence.

Our chosen descriptors of natural gesture can be summarized as three basic objectives: (1) The generator should produce sequences of joint positions that represent valid human skeleton configurations. (2) The produced pose sequences should describe realistic gesture dynamics, including distinct phases such as periods of acceleration as well as

stillness. (3) The output pose sequences should be appropriate with respect to the speech they accompany. With this selection of objectives, we aimed to ensure that our output can both be considered speech gesture (valid human skeleton moving according to speech), as well as addressing the problems of previous works of overly smooth or lethargic motion by explicitly enforcing some characteristics of gesture motion dynamics.

In the next Section, we will discuss how we represented the above output objectives with a set of training adversaries, called discriminators, each enforcing a different part of the objectives. Each discriminator was a separate neural network, with its own training loss feedback. Their architectures are detailed in Figure 5.8; we will describe each discriminator one-by-one below.

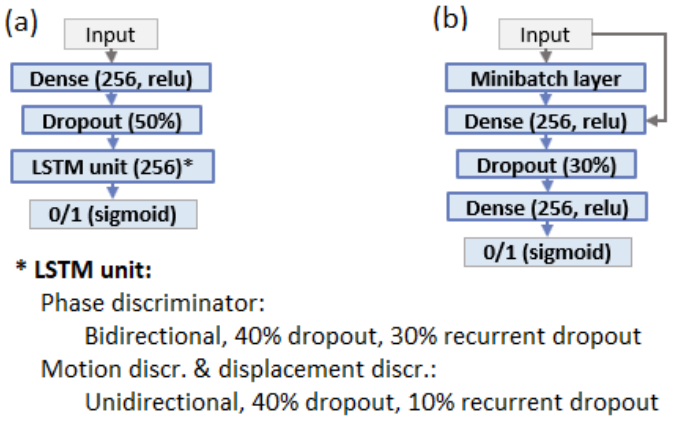


FIGURE 5.8: Network architecture of the adversaries. Left: Phase, motion, and displacement discriminators. Right: Minibatch discriminator. All discriminators applied input transformation via a standard densely connected NN layer. (The minibatch layer applied Equation 5.3 before the input transformation.) Dropout was applied subsequently, followed by a recurrent unit (left) or another densely connected NN layer (right). The output layer applied a sigmoid activation.

5.5.1 Phase structure discriminator

The phase discriminator's job was to determine whether the generator's output follows a realistic gesture phase structure. This discriminator therefore only received phase labels as input rather than joint positions. We additionally provided the phase discriminator with the pitch value at each time-step as an indicator of speech emphasis. The network architecture of the phase discriminator is detailed in Figure 5.8a.

Phase labels were always determined by the phase classifier; that is, we never used the ground truth annotation during adversarial training. This ensured that any differences

in the phase structure of real and generated data was not due to potentially noisy automatic classification. As the phase labels were automatically determined by the phase classifier, we wanted to ensure somewhat sensible input to the classifier, i.e. input resembling human motion, and we therefore pre-trained the generator with a standard MSE, allowing fast convergence to a skeleton structure (see Section 5.4.2). Before adversarial training, we similarly prepared the phase discriminator, essentially letting it "catch up" to the pre-trained generator. For this, we utilized the training history of the generator's pre-training: The generator's network weights saved periodically during its pre-training. For pre-training the phase discriminator we used this history as follows: The phase discriminator received the classified phase labelling of an untrained generator (i.e. noise input); when the phase discriminator achieved an accuracy score of at least 70% for three batches in a row, the generator got 'upgraded' with the next set of weights from the training history. This was repeated until the phase discriminator had reached the level of the fully pre-trained generator (last saved set of network weights). This step-by-step upgrading of the generator's weights served to not overwhelm the discriminator during pre-training.

5.5.2 Motion realism discriminator

Adversarial training between the generator and the phase discriminator alone would quickly lead to divergence from the skeleton structure due to the phase discriminator only judging the automatically classified phase labels. As described in Section 5.4.2, the phase classifier may assign a realistic phase structure to unrealistic input; when the generator is judged solely on this phase structure, it may receive positive discriminator feedback for entirely unrealistic output and we found this to lead to increasing divergence from skeleton-like joint positions. To address this problem, we employed a second discriminator that judges the output of the generator directly by receiving the raw generated joint positions, as well as the corresponding audio features. This discriminator received as input the 63 joint values (x, y, z of 21 joints) and 27 speech features. Its network architecture is detailed in Figure 5.8a.

The motion realism discriminator was pre-trained in a classic adversarial training setting with a new generator in order to learn to detect unrealistic point clouds not resembling

a skeleton. This was necessary in order to not allow the already pre-trained generator to regress to non-humanoid point clouds.

5.5.3 Minibatch discriminator

Adversarial training is prone to suffering from mode collapse, where the generator produces repetitive patterns of output. While the discriminator can immediately learn that this specific pattern comes from the generator, the generator only needs to shift its repetitive output slightly to fool the discriminator. This may be repeated in an infinite cat and mouse game. One reason for this mode collapse is that a standard discriminator only judges one output sequence at a time, rather than in the context of a whole batch of data. A minibatch layer can be added to allow the discriminator to see this context and ensure that the generator cannot get away with even novel patterns when they are repetitive throughout the data batch [157].

Instead of integrating minibatch discrimination into the motion realism discriminator, we achieved better performance when outsourcing the task to a separate discriminator. This discriminator received the 63 joint values (x,y,z of 21 joints) generated by the generator or taken from the ground truth and calculated a minibatch similarity measure:

$$sim(X) = L^1(W \cdot X), \tag{5.3}$$

where L^1 denotes the L1 norm and W is a 300-dimensional (trainable) weight tensor. The detailed architecture of the minibatch discriminator is shown in Figure 5.8b.

5.5.4 Displacement discriminator

The generator's output at the beginning of adversarial training was the damped motion learned from the MSE pre-training. To encourage the generator towards less damped motion, we introduced a displacement discriminator that received the same motion input as the phase classifier, namely the per-frame x, y, and z offset of the 16 arm joints (48 values). That is, the displacement discriminator explicitly saw how much each joint has moved at each time-step; it could penalize a generator that produces very slow (or very fast) motion. In effect, the displacement discriminator judged the directional velocity of the generated joint positions. The displacement discriminator also served to reduce

jitter in the motion (offset in one direction always followed by some offset to opposite direction).

The error from this discriminator received a lesser weight and serves as a minor side objective of the generator training, helping to stabilize and speed up convergence and smooth output motion. The architecture of the displacement discriminator followed that of the motion realism discriminator and is visualized in Figure 5.8a.

5.6 Training process

During adversarial training, the generator's output was judged by all discriminators and an averaged error was computed, as detailed in Section 5.6.1 below. This was followed by a training step of objective numerical errors. The objective error functions sped up convergence and enabled continuous prediction, as described in Section 5.6.2.

5.6.1 Adversarial training

The adversarial training is visualized in Figure 5.9 and summarized below:

- The **generator** receives 27 prosodic speech features as input and generates corresponding 3D positions of 21 joints.
- The **phase classifier** first converts the joint positions to frame offsets and subsequently predicts a sequence of gesture phase labels. The phase classifier also receives as input the F0 (pitch) value of each frame. The classifier's weights are fixed during adversarial training.
- The produced phase label sequence of the classifier, plus the F0 value, serve as input for the **phase structure discriminator**.
- The **motion realism discriminator** receives the joint positions directly, as well as all corresponding 27 speech features.
- The **displacement discriminator** receives the same motion input as the phase classifier, the per-frame joint offsets of the 16 arm and hand joints.
- The **minibatch discriminator** only receives the joint positions as input.

All three discriminators were trained with a binary cross-entropy loss to determine whether a motion sequence is real or generated. The discriminators learn independently from each other, sharing no weights and receiving individual training loss feedback. The loss of the generator with respect to the three discriminators was weighted and combined into a single value for the generator's training step. All models worked with input sequences of 5 seconds, at 20 fps, resulting in 100 time-steps.

During adversarial training steps, the generator optimized the binary cross-entropy of the discriminators' output. The generator's training error with respect to the four discriminators was averaged for each optimization step in the following manner:

$$\mathcal{L}_{GAN}(G) = \frac{w_p\mathcal{L}(G, D_p) + w_r\mathcal{L}(G, D_r) + w_m\mathcal{L}(G, D_m) + w_d\mathcal{L}(G, D_d)}{w_p + w_r + w_m + w_d},$$
$$\text{with } w_p = 2, w_r = 4, w_m = 4, \text{ and } w_d = 1,$$

where w_p is the weight assigned to the phase discriminator's loss, w_r the weight for the motion realism discriminator, w_m the weight for the minibatch discriminator, and w_d the weight for the displacement discriminator. $\mathcal{L}(G, D)$ represents the generator's objective with respect to one discriminator. The weighting of 2:4:4:1 was chosen by empirically finding values that led to stable training with respect to all discriminator objectives, without the generator collapsing with respect to one or more objectives. The adversarial training of the generator is visualized in Figure 5.9, representing a more detailed version of the previously presented Figure 5.2. We used the RMSprop optimizer during adversarial training.

5.6.2 Objective loss penalties

In addition to the adversarial updates of the generator, one MSE correction was performed per two adversarial steps. The MSE avoids major deviations of the generator's output from a realistic skeleton structure that would produce nonsensical phase label output and slow down the training overall. An alternative, similar approach would be to restrict joint positions to realistic ranges.

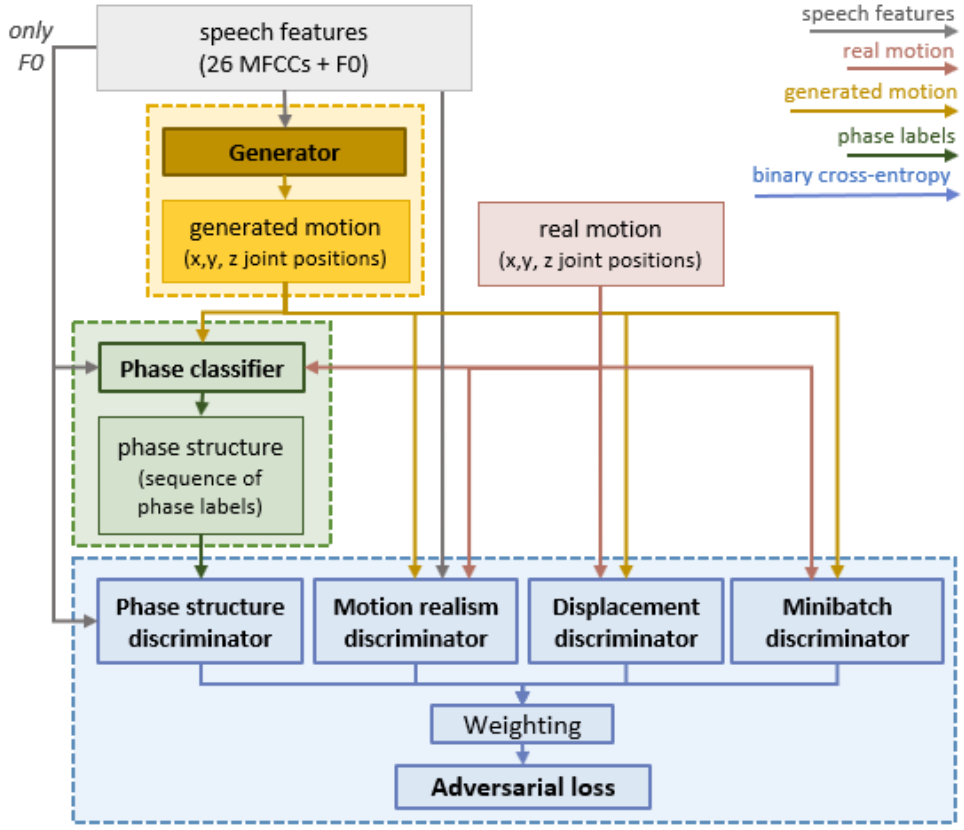


FIGURE 5.9: Adversarial training. The generator produced joint positions based on input speech features. Its output was judged by four discriminators with separate objectives, and a weighted error was computed with respect to all four evaluations. Each discriminator optimized the binary cross-entropy objective, deciding if a given data sample is real or generated.

The generator was trained to predict gesture motion for 5 seconds of speech input at a time rather than for continuous input. Gesture motion was therefore continuous within 5 second prediction intervals, but could be visibly discontinuous between intervals. To avoid having to compute smooth transitions in post-processing, we introduced a penalty for the generator for discontinuous sequences within a training batch. The discontinuation penalty was computed as the mean squared distance between the start position of a sequence and the end position of the preceding sequence. The penalty for first sequence within a batch was always set to zero and otherwise:

$$\mathcal{L}_{cont}(G) = \frac{1}{T}\sum_{t=1}^{T}(G(x)(t) - G(x)(t-1)))^2 \ . \tag{5.4}$$

We observed during adversarial training that the predicted finger positions often moved far from the hand. To speed up the training process, we added a simple finger distance

penalty restricting the predictions to realistic ranges. We computed the distance of each finger marker to the respective hand marker and calculated the MSE with respect to the real distances:

$$\mathcal{L}_{fingers}(G) = \frac{1}{n}\sum_{i=1}^{n}(\mathcal{D}_{fingers}(G(x)) - \mathcal{D}_{fingers}(Y(x)))^2 \tag{5.5}$$

with $Y(x)$ denoting the ground truth for sample x, and $\mathcal{D}_{fingers}$ computed as the concatenation of each finger marker's x, y, and z distance from the respective hand.

5.7 Results

We conducted a series of qualitative evaluations to clarify the roles of each discriminator and their benefits for generator training, and quantitative evaluations of the resulting generator output. For our qualitative evaluation, we assess the output visually and discuss the subjective performance. We will below refer to an illustrative video at `https://youtu.be/tHKqDPy8vHU`.

5.7.1 Qualitative evaluation

5.7.1.1 Phase structure discriminator

The phase structure discriminator allowed us to capture important gesture dynamics without having to rely on implicit learning from a larger dataset (such as in Ginosar et al. [124]). During the pre-training described in Section 5.5.1, this discriminator easily learned to distinguish the (noisy) classified phase structures of real motion and motion produced by the pre-trained generator. During adversarial training, the phase discriminator's accuracy remained balanced with the generator's while the generator's output was improving in quality. We visualize the benefits of the phase discriminator for encouraging better gesture motion dynamics in a video at `https://youtu.be/tHKqDPy8vHU`; without the phase discriminator, the motion showed no clear holds or accelerations characteristic of the stroke phase. The motion appeared to correspond less with the speech prosody.

5.7.1.2 Motion realism discriminator

The phase discriminator's judgment alone was not a sufficient constraint for the generator's output. As described in Section 5.5.2, the automatic phase label classification of the generator's output and the phase classifier's naivety with respect to non-human point clouds provided too much room for the generator to produce unrealistic data. The motion discriminator presented a better constraint for maintaining a skeleton structure as it saw the generator's output directly and successfully constrained the generator to data points resembling a skeleton structure. Figure 5.7e visualizes the output distribution produced by a generator unconstrained by a motion discriminator. The video at `https://youtu.be/tHKqDPy8vHU` also shows a sample of the motion produced without a motion realism discriminator; the joint positions move away from the skeleton structure, producing output not resembling human motion.

5.7.1.3 Minibatch discriminator

As a vanilla discriminator only judges output sequences in isolation, without taking the context of the data batch into consideration, the generator can suffer from mode collapse, as described in 5.5.3, and visualized by the plotted data distribution in Figure 5.7c. Our minibatch discriminator successfully forced the generator to produce more diverse output. The video at `https://youtu.be/tHKqDPy8vHU` shows the repetitive motion generated under mode collapse, as well as the improved, more diverse output with minibatch discrimination. We considered two alternative integrations of minibatch discrimination into our model, namely as part of the motion realism discriminator and as part of a separate discriminator. In practice, we found adversarial training to be more stable when outsourcing the minibatch discrimination to a separate discriminator only receiving motion input. Generator training was less likely to collapse with respect to one discriminator when the adversarial objective was more distributed. The benefit of employing multiple discriminators has also been discussed in previous works [158, 159].

5.7.1.4 Displacement discriminator

Learning from the phase discriminator's feedback would potentially be difficult for the generator due to the hidden layers between the generator and phase discriminator (i.e.,

the phase classifier's computations that are inaccessible to the generator). The generator's motion output was first converted to per-frame offsets of the joints and then passed to the classifier for higher level feature extraction. Introducing a discriminator receiving the same processed motion as the classifier can provide more direct feedback. In practice, we found that the addition of such a displacement discriminator sped up learning and moved predictions away faster from the damped baseline motion produced by the pre-trained generator. We visualize this by plotting an example data distribution in Figure 5.7d. The slow departure from the mean pose when training the model without the displacement discriminator is also shown in the video at `https://youtu.be/tHKqDPy8vHU`. We also illustrate the smoothing benefit of the displacement discriminator in the video: When training the generator without any discriminator receiving the joint offsets (i.e. with neither the displacement discriminator nor the phase classifier and discriminator), the motion output displayed a great amount of jitter. We show that adding the displacement discriminator reduced jitter to a large degree. This discriminator received the smallest weighting in the generator's objective.

5.7.1.5 Adversarial error weighting

We found a weighting of 2:4:4:1 for the error of the phase discriminator, motion realism discriminator, minibatch discriminator, and the displacement discriminator, respectively, to achieve the most stable training, measured by the accuracy of the binary cross-entropy objective for each discriminator. This weighting allowed us to see stable accuracy improvements for the generator across all adversarial objectives without collapse with regard to one or more objectives.

5.7.1.6 Objective losses

The discontinuation penalty was largely successful in reducing the positional jumps between predicted motion sequences, making the model more applicable for continuous gesture generation for long sequences of speech input. The finger distance penalty proved a simple measure to avoid unrealistic finger positions without strongly constraining the generator in its predictions.

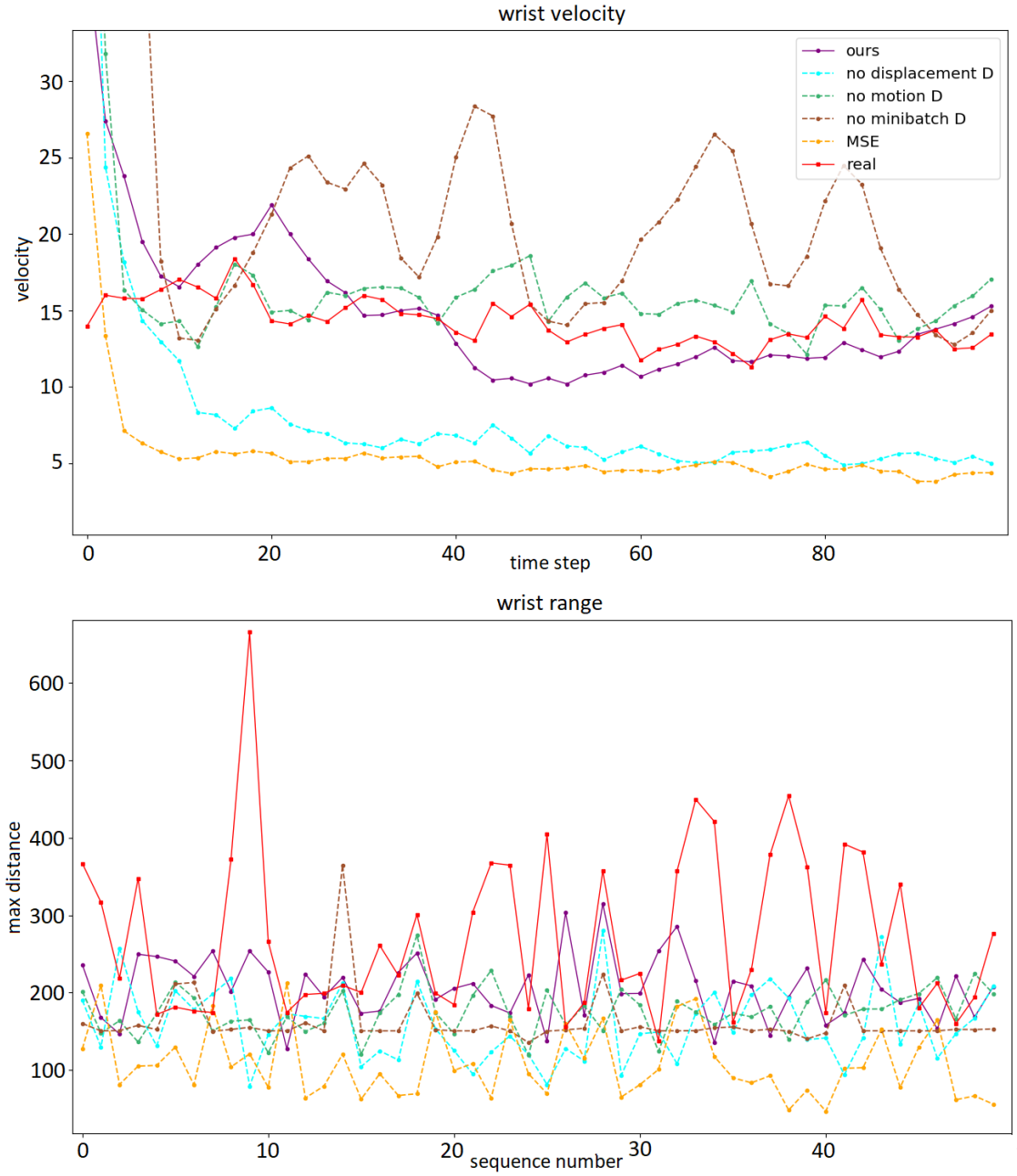


FIGURE 5.10: Quantitative gesture generation evaluation. Top: Wrist velocity for each predicted time step, median across 150 sequences (see Equations 5.6 and 5.7). Bottom: Maximum distance of the wrists from mean pose for 50 randomly selected sequences.

5.7.2 Quantitative evaluation

We provide a quantitative evaluation of our generation results based on the wrist motion in Figure 5.10. We present these results in an ablation manner, as in Figure 5.7, evaluating how removal of a specific discriminator in training affects the generation result.

The top graph plots the wrist velocity per predicted time step, each representing the

median over 150 predicted gesture sequences. This 1-dimensional velocity of the 3-dimensional x, y, z joint coordinates of a time step t and a sequence i is more specifically calculated as follows:

$$velocity(t^i) = |x_{t^i} - x_{t^{i-1}}| + |y_{t^i} - y_{t^{i-1}}| - |z_{t^i} - z_{t^{i-1}}| \tag{5.6}$$

$$velocity(t) = median(t^0, t^1, ...t^i, ...t^n) \tag{5.7}$$

We can see that one of the closest matches of real motion (red) are achieved by our model (purple) and the system configuration removing the motion discriminator (green). However, the latter configuration generated joint positions that heavily violated human skeleton constraints. Removing the minibatch discriminator (brown) produced faster than real motion, as well as resulting in highly repetitive output. The output under removal of the displacement discriminator (blue) as well as the output the generator trained solely with a mean squared error loss (yellow) exhibited very slow motion, much below realistic levels.

The bottom graph in Figure 5.10 plots the maximum distance travelled away from the mean pose, for 50 example sequences. The closest match to real wrist position ranges was achieved by our model, though it does not reach the wide ranges of real motion. The MSE-trained generator and the no-displacement-discriminator condition showed a comparable level of variation to real motion, but the gestures were overall closer to the body both than real motion and than for our model. The no-motion-discriminator condition similarly produced lower ranges than real motion. The no-minibatch-discriminator condition produced very stable ranges, indicative of the repetitive gesture sequences generated.

5.8 Discussion

We explored generative adversarial networks for speech-to-gesture translation with higher level feature extraction. Gesture motion is marked by distinct dynamics, including phases of acceleration and effort, of pause, and of relaxation. These higher-level dynamics can be difficult to capture implicitly. To enforce these dynamics more explicitly in a top-down manner, we train a classifier to detect gesture phases automatically, and

then train a phase structure discriminator to detect realistic versus non-realistic phase sequences.

To train the phase classifier, we hand-annotated the phases of an over 3.7 hour long subset of dataset 2 using 9 different phase labels. We validated our results on a different speaker, for whom we annotate an additional small sample of gesture sequences. We compared three models of phase classification with different levels of detail (1-, 4-, and 6-class classification). We achieved good results, and we conclude that our error rate may, to a relatively large extent, be due to inter-coder inconsistencies. This leads to the dilemma of weighing data quantity against data quality; the large time requirement of hand-annotation (1 hour or more work for 1 minute of data) tempts distributing the work load across a number of people, but this may lead to increased problems with annotation consistency. When motion capture is available, we suggest that automatically pre-annotating all sections with close to zero velocity as 'hold' could speed up the annotation process as well as increase inter-coder agreement in future work.

Our 1-class stroke classifier performed similarly well on both the training and the validation speaker. 4- and 6-class classification reached equal scores for the training speaker; for the validation speaker, the 4-class model achieved a significantly higher score. One reason for the drop in performance on the validation speaker for the multi-phase models may be differences in speaker style, leading to different expressions of gesture phase. The higher the level of detail, the larger are the expected inter-speaker differences. Ideal phase classification may therefore always be speaker-specific.

For training the gesture generator, instead of using a standard regression loss, we constructed a generative adversarial setting with multiple discriminators. We observed a clear advantage of adversarial training over using a standard regression loss; the produced motion had a larger positional range, more realistic velocity, and appears much less damped.

By using multiple discriminators, we could phrase the speech-to-gesture generation problem as a series of sub-problems. We used our automatic phase labelling to enforce a more realistic gesture phase structure in our output; this was the task of the phase structure discriminator. The phase structure discriminator enabled the enforcement of higher level dynamic characteristics in the output without having to rely on implicit learning from a large amount of data.

Because an automatic phase classifier would always assign some phase label to even random point clouds, we constrained the motion output with a second discriminator judging the generated joint positions as real or fake; this was the task of the motion realism discriminator. Because the motion realism discriminator's task was to judge one generated motion sequence at a time, it could allow for the same sequence to be generated repeatedly. A minibatch discriminator detected such repetitive patterns, ensuring diversity in the output. Lastly, generated motion can often look jittery; we addressed this by including a the training objective of realistic joint displacement per frame, monitored by the displacement discriminator.

To our knowledge, this is the first work using adversarial training for generating 3D gesture motion from natural speech, and the first work exploring the use of multiple discriminators for the purpose. We observed a benefit of using multiple discriminators to stabilize adversarial training, and we reported how each discriminator addresses a distinct sub-problem in the gesture generation task. We employed explicit modelling of the dynamics of gesture motion to allow learning of these higher level features from a smaller dataset. We see our work as a further step towards enabling automatic animation of realistic conversational agents.

Our results are limited to gesture generation for the single speaker we utilized and more data of various speakers would be necessary to make generalizations. Due to the high variance of gesture behavior across speakers, this is a very difficult task. Because we generated gesture motion from prosodic speech features, semantically meaningful gestures can hardly be inferred without explicitly employing speech recognition methods. Speech recognition, however, would likely only yield a benefit when using a much larger dataset, ensuring a number of examples of the same phrases.

While generated motion improved greatly with respect to standard regression loss training, the produced motion still lacked desirable levels of realism. As we have discussed previously in this thesis, gesture motion and its relationship to speech is quite complex, and generating realistic gesture based on speech is therefore a difficult task to train requiring guiding feedback to the generator. While in the previous chapter we have discussed that a standard training function such as the mean squared error on joint positions or angles is too strict and constraining for the generator, the very simple feedback of a discriminator's believe/ don't believe judgment may be too little information

for the generator to find a good solution, more "teacher guidance" may be beneficial. Looking forward, in the next chapter, we will evaluate other measures of gesture appropriateness. Using the gesture phase extraction of this chapter, we will conduct a deeper analysis of the relationship between gesture characteristics and accompanying speech by investigating more specific descriptors of a gesture that may be used for assessing a gesture model output.

Chapter 6

Gesture Parameters from Speech

In this chapter, we explore gesture representation through higher level motion parameters.[1] We propose five gesture parameters and assess how well they may be modelled from speech as well as their impact on perceived speech-gesture match.

6.1 Introduction

We have discussed that a major difficulty in modelling the speech-to-gesture relationship is the highly non-deterministic input-to-output relation. There are a multitude of possible gestures for each utterance, varying not only between speakers but also within. Therefore modelling gestures as sequences of joint positions or angles can fail to capture the natural variety of gesture motion. We have explored the use of a generative adversarial training paradigm avoiding any explicit feedback with regard to joint pose, however, this appeared to lack learning information for the generating network. In this chapter, we explore alternative representations of gesture that do not rely on explicit joint positions or angles while allowing more specific modelling of the gesture motion. We investigate a number of gesture descriptors for their usability in this task, specifically, by assessing their computability from a speech signal as well as their impact on perceived gesture performance.

[1] The contents of this chapter were published at the ACM International Conference on Intelligent Virtual Agents 2020 (IVA'20) (Listed Publication 4). The second author, Michael Neff, acted as an additional advisor, contributing to the study design, and provided the animation engine.

In light of the discussed difficulties designing a speech-to-gesture system, we want to determine, if not through explicit joint poses, how *can* we infer gesture information from speech? The *Growth Point* theory of McNeill [15] suggests that rather than speech directly informing the gestures to be produced, speech and gesture are both expressions of the same cognitive process, two channels expressing the same idea. Therefore, speech may give us an indication of the underlying intention that inspired a gesture, but may never fully predict the gesture expression. Often however, we want to rely solely on the speech signal for generating gesture behavior due the ease of obtaining such speech in real applications. For generating gestures from a speech signal, we are interested to what extent we *can* predict the expressive qualities of gesture from speech; specifically, which characteristics of gesture correlate well with the speech signal, can be predicted successfully and are perceptually important. To this aim, we first need to determine a number of gesture parameters, that describe the expressiveness of a gesture.

The previous work presented on parameter representation of gestures in Section 2.6.2 shows that we can reliably influence perceptions of personality by applying simple modifications, and gives some evidence that matching measures of gesture expressivity to speech can increase appeal. While tackling the speech-to-gesture problem, we are interested in which gesture parameters are related to the speech expression. On the one hand, we would like to know which gesture parameters can be successfully predicted from speech. On the other hand, we want to understand which of these parameters are important for perceived plausible gesture synthesis. In this chapter, we build upon both previous work on modelling the speech to gesture relationship, as well as work on gesture representation through parametrization of the motion expressivity. The previously discussed work by Hartmann et al. [94] (see Section 2.6.2) reported evidence that matching expressive motion parameters, such as scale, fluidity, and dynamic power, to the communicative intent makes the gesture behavior more appealing.

Firstly, by training multiple recurrent networks to model the speech to gesture parameter relationship, we could assess how well a particular gesture parameter may be inferred from a speech signal. Secondly, we determined the perceptual relevance of the gesture parameters in an empirical study. Examining the perceptual saliency of attributes of gesture motion provides guidance on what features must be accurately modeled to produce satisfying results.

6.2 Data processing

We used the two datasets presented in Chapter 3, combined representing over 10 hours of synchronized high-quality speech and motion recordings. In the sections below, we describe how speech and motion data was processed to extract feature sets.

6.2.1 Speech processing

We tested the suitability of three different feature sets for speech processing. The first set consists of the 12 Mel-frequency cepstral coefficients (MFCCs), common in speech recognition as well as previous speech-gesture work [119, 160]. Secondly, we tested Geneva Minimalistic Acoustic Parameter Set (GeMAPS), both the 18 features of the compact version, as well as an extended set of 23 features presented in Eyben et al. [161]. The GeMAPS has been specifically developed for affect recognition. Finally, we tested a three feature set simply consisting of the pitch (F0), plus its first and second derivative to describe change over time. We extracted all speech features using openSMILE [151]. After training a number of speech-to-gesture-parameter models in an exploratory manner with each of the three feature sets, we found GeMAPS to work best overall, as measured by the numeric loss during training, with the compact and the extended feature set performing similarly. MFCCs performed well but slightly worse than GeMAPS, and the feature set of pitch plus derivatives greatly underperformed. We will therefore report results using the GeMAPS input representation.

6.2.2 Gesture processing

We aimed to find a number of gesture characteristics that could describe the expression of a gesture. We define these characteristics based on the central part of a gesture, the stroke phase, which represents the expressive phase of a gesture and carries its meaning [15]. The stroke phase was determined following the approach of Chapter 5, using the hand-annotation where available, and the automatic stroke classification otherwise. We extracted gesture characteristics with a custom animation tool using motion parametrization similar to Neff and Kim [162] and based on DANCE [163] to automatically calculate the features of interest. Input is given as a motion capture file

and a list of timestamps indicating the location of the stroke phases. The five chose gesture characteristics are listed below and detailed in Equations 6.1-6.5.

1. velocity
2. initial acceleration
3. gesture size
4. arm swivel
5. hand opening

Velocity and initial acceleration both describe the kinematics of the gesture, represented by the maximum stroke velocity (1), and by the mean acceleration to the first major velocity peak (2). Velocity captures a character's tempo and relates to the amount of energy they are using. Initial acceleration may be useful to model an emphatic gesture start. This is akin to the type of tangent adjustment between keys in hand animation.

With (3), gesture size described the spatial extent of the gesture. We measured this in two ways: The total path length of the gesture stroke, and length between the minimum and maximum point of the stroke, which we will subsequently refer to as major axis length.

Parameter (4), arm swivel, describes the elbow angle, the distance between the elbows and the torso. This angle modifies the amount of space taken up by the gesture and can change the perceived personality [81] and has been postulated to relate to humility and arrogance [164].

The last parameter, (5), describes the hand shape during a gesture, specifically, how open or closed the hand is. We calculated this as the mean distance of the the finger tips (excluding thumb) from the base of the wrist. Such variation in hand flexion has been shown to impact the perception of character personality [83].

Based on previous work, we expected gesture velocity and acceleration to be well predictable from speech (e.g. [165, 166]), whereas more uncertainty surrounded speech correspondence to arm swivel and hand opening.

The average values for these gesture parameters are listed in Table 6.1, separated by speaker. We can see clear differences between the two speakers; for example, speaker 1 shows a higher maximum stroke velocity as well as much higher initial acceleration.

For modelling gesture parameters from speech in the next section, we did not explicitly include speaker identity, however, speaker identity was implicitly given through prosodic speech features.

EQUATIONS 6.1 - 6.5: Calculations of gesture parameters

$$velocity_{max} = max(\frac{\Delta x}{\Delta t} + \frac{\Delta y}{\Delta t} + \frac{\Delta z}{\Delta t}) \tag{6.1}$$

$$\begin{aligned} initial\ acceleration &= mean(velocity[0 : peaks[0]])\ , \\ \text{where } peaks &= max_{local}(velocity) \\ \text{and } min.\ peak\ distance &= \frac{duration}{3} \end{aligned} \tag{6.2}$$

$$path\ length = \Delta x + \Delta y + \Delta z \tag{6.3.1}$$

$$\begin{aligned} major\ axis\ length &= p_{max} - p_{min}\ , \\ \text{where point } p_{min} &= argmin(p_x + p_y + p_z) \\ \text{and point } p_{max} &= argmax(p_x + p_y + p_z) \end{aligned} \tag{6.3.2}$$

$$\begin{aligned} arm\ swivel &= \alpha(plane_{arm}, plane_{shoulder,wrist})\ , \\ \text{where } plane_{arm} &\text{ is the plane passing through shoulder, elbow, and wrist,} \\ \text{and } plane_{shoulder,wrist} &\text{ is a vertically aligned plane passing through shoulder and wrist} \end{aligned} \tag{6.4}$$

$$hand\ opening = \frac{\sum_{i=2}^{5} f_{i_{xyz}} - handbase_{xyz}}{4} \tag{6.5}$$

TABLE 6.1: Average gesture parameter values for the 2 speakers. Note that for arm swivel, increasing the swivel angle (moving the elbow further out) means higher positive values for the left arm, but increasingly negative values for the right arm.

	Left hand		Right hand	
	speaker 1	speaker 2	speaker 1	speaker 2
velocity	0.81	0.43	0.90	0.55
initial acceleration	0.40	0.06	0.49	0.07
path length	0.33	0.21	0.38	0.29
major axis length	0.18	0.13	0.2	0.18
arm swivel	17.51	12.85	-22.10	-20.75
hand opening	19.10	14.75	19.17	14.90

6.3 Gesture parameter prediction

The first part of our work focused on the problem of predicting gesture characteristics from a speech signal. Our aim hereby was to assess which gesture descriptors can be predicted from speech with the current machine learning techniques. In an exploratory manner, we trained a large number of input-output combinations, empirically determining suitable model parameters.

We utilized recurrent neural networks due to their strength in modelling sequential time-series data. All models took an input sequence of speech features, extracted over the period of the corresponding gesture's stroke phase plus a context of 1 second in each direction. Sequence-based models require a constant input length, we therefore defined a maximum input length of 5.5 seconds, based on the maximum stroke duration found in the datasets, including context windows. All shorter sequences were zero-padded to fulfill the constant input length requirement.

The general network structure that performed best overall is shown in Figure 6.1. Using more recurrent layers or a larger recurrent layer size led to frequent over-fitting; using smaller recurrent layer size or a uni-directional recurrent layer led to under-fitting.

The model applied batch normalization to the input and input transformation through a feed-forward layer of size 64. The transformed input was passed through one or two recurrent network layers, followed by batch normalization and a dropout

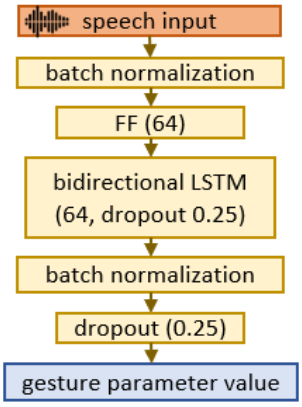


FIGURE 6.1: Network structure of the speech-to-gesture-parameter models. Speech input was batch-normalized, then passed through a linear feed-forward layer of size 64. The core of the model was a bidirectional LSTM cell of size 64 (25% input dropout). The output of the recurrent cell was batch-normalized and 25% dropout was applied before the final output layer with sigmoid activation.

TABLE 6.2: Performance evaluation of the speech-to-gesture-parameter models. In brackets are the random sampling error, computed using a randomly drawn a parameter sample from all true values as prediction value. We report mean ($\bar{e}$)and median ($\tilde{e}$) errors for the left (L) and right (R) hand, as well as the % reduction in error between random sampling and our models.

	$\bar{e}$ L	red.	$\bar{e}$ R	red.	$\tilde{e}$ L	red.	$\tilde{e}$ R	red.
velocity	0.32 (0.44)	28%	0.35 (0.49)	30%	0.26 (0.33)	23%	0.29 (0.38)	24%
initial acc.	0.28 (0.42)	34%	0.34 (0.48)	29%	0.12 (0.19)	38%	0.14 (0.23)	40%
path length	0.14 (0.28)	50%	0.15 (0.33)	55%	0.08 (0.17)	53%	0.09 (0.21)	57%
maj. ax. len.	0.09 (0.14)	35%	0.09 (0.16)	46%	0.05 (0.10)	47%	0.05 (0.12)	57%
arm swivel	11.44 (16.78)	32%	9.57 (14.72)	35%	8.54 (12.29)	31%	6.96 (11.3)	38%
hand opening	1.45 (2.30)	37%	1.38 (1.88)	27%	0.96 (1.51)	36%	0.99 (1.27)	22%

layer for regularization purposes. The outputs of a model were the values of the gesture parameter under investigation (e.g., velocity, initial acceleration, etc.), normalized to the range of 0-1 for each given stroke, either a single value for both hands or one for each, as described below. The output nodes had a sigmoid activation. Training minimized the mean squared error between predicted and true value.

To generate output, it is necessary to resolve a potential ambiguity between the predicted behavior of each hand. The stroke label does not include the handedness, i.e. whether the right, the left, or both hands are performing a stroke. Therefore, the predictive model must make some assumptions about the active hand(s). We considered modelling a single output value defined as the maximum of both hands instead of making predictions for each hand, but did not find this to improve results. The second option is outputting two values, representing the two hands. The difficulty in predicting two values can be that the model has to infer the gesture handedness for accurate prediction. The model can learn general statistics regarding differences between the two hands (e.g. left hand generally slower), but will not be able to predict diverging values indicating gesture handedness (e.g. high velocity for right hand and zero velocity for left hand, indicating a right-handed gesture).

6.3.1 Model training

Using the stroke phases as our segmentation, our training data consisted of a total of almost 23,700 gesture stroke samples, with approximately 42% from the dataset 1 (see 3.1) and 58% stemming from dataset 2 (see 3.2). We held back about 5% of the samples

for validation, chosen randomly. The velocity and acceleration models reached best performance after 70 epochs, all other models were trained for about 140 epochs.

For each model input, we tested the compact GeMAPS with 18 speech features versus the extended GeMAPS of 23 features performed. The compact and the extended set performed very similarly, and we will report best model results in each case. The reported models used the compact set in the case of gesture size, and the extended set in all other cases.

6.3.2 Results

Below, we will report prediction results for all gesture parameter models. In order to evaluate the model performances, we compare prediction errors to random sampling errors in Table 6.2. In each case, we computed the error as the difference between the predicted and actual value for each gesture in our test data, and then averaged over all gestures. Random sampling errors were computed by using randomly drawn samples from the entire true data set as prediction values. This ensured that the random samples follow the true distribution of the dataset. We also drew the samples in a database-specific manner, i.e. we always used the correct database to draw from for each sample, to ensure that simple speaker-detection by the model would not be the reason for its superior performance. We computed the random sampling error three times for each gesture parameter and report the average values. The random sampling error represented our error baseline; models that performed better than this baseline detected some relationship between input speech and output gesture value.

6.3.2.1 Gesture kinematics

We found the gesture kinematics, as described by the maximum velocity and the initial acceleration mean, relatively difficult to model, and compare below the use of a prediction for each hand or a combined prediction, as described in the previous section. The **maximum velocity** across both hands averaged 0.80 m/s ($std = 0.52$ m/s) and the best model resulted in a mean error of 0.35 m/s ($median = 0.27$ m/s) when compared to the ground truth gesture data. The model avoided very low velocity predictions, as

well as to some degree high velocity predictions (visible in Figure 6.2 (top)). For modelling two hands, the velocity measure averaged 0.60 m/s ($std = 0.46$ m/s) for the left, and 0.70 m/s ($std = 0.49$ m/s) for the right hand, and our model produced mean errors of 0.32 m/s and 0.35 m/s respectively, with the median at 0.26 m/s and 0.29 m/s. Referring to Table 6.2, we see 28% mean error reduction for the left, and 30% for the right hand, compared to random sampling, and 23% and 24% median error reduction.

In the case of modelling the maximum value across both hands, **initial acceleration** averaged 0.36 m/s^2 ($std = 0.72$ m/s^2) and the model produced a mean error of 0.36 m/s^2 ($median = 0.15$ m/s^2). Modelling hands individually, the acceleration measure averaged 0.13 m/s^2 ($std = 0.54$ m/s^2) for the left, and 0.27 m/s^2 ($std = 0.66$ m/s^2) for the right hand, and our model produced mean errors of 0.28 m/s^2 ($median = 0.12$ m/s^2) and 0.34 m/s^2 ($median = 0.14$ m/s^2), respectively. The model again avoided very high acceleration predictions, however, high acceleration is often correctly identified though the predicted value tended to be lower than the true value (see plotted prediction results in Figure 6.2 (bottom)). Compared to our baseline random sampling error, we achieved a mean error reduction of 34% and 29% for the left and right hand, respectively, and 38% and 40% median error reduction (see Table 6.2).

6.3.2.2 Gesture size

Our first measure of gesture size was **path length**. We found that gesture path length was highly correlated with the length of the corresponding input speech segment; a longer speech input was associated with a longer stroke. Hence, in addition to comparing prediction results to the random sampling error (see Table 6.2), we employed a second test taking into account speech length. We trained a control model only on speech length, that is, the single input speech feature has the value 1 for all input time steps before the zero-padding. This input processing meant that the model could base predictions solely on the length of the input signal.

Using only speech length versus GeMAPS input yielded very similar errors: With mean path lengths of 0.26 m ($std = 0.30$ m) and 0.33 m ($std = 0.35$ m) for the left and right hand path length, respectively, using only speech length input yielded mean errors of 0.14 m ($median = 0.07$ m) and 0.16 m ($median = 0.07$ m) for the left and right hand, while using GeMAPS resulted in mean errors of 0.14 m ($median = 0.08$ m) and 0.15

m ($median = 0.09\ m$), respectively. However, relying only on the correlation between input and output length failed to predict larger path lengths (for visual comparison refer to Figure 6.3). Paired t-tests showed that our model using GeMAPS predicted path length for the right hand significantly better ($p < .001$, left: $p = 0.13$).

Our second measure of gesture size was the **major axis length**, defined as the length of the axis between the minimum and maximum point of the gesture. The average major axis lengths for the left and right hand were 0.16 m ($std = 0.15$) and 0.18 m ($std = 0.14$), respectively, and our model produced mean errors of 0.09 ($median = 0.05$) and 0.09 ($median = 0.05$) for the left and right hand, respectively (see also Figure 6.4). We critically evaluated the results for the major axis length in the same manner as for the path length, using only speech length as input. Similarly as with path length, when using solely speech length information, the model often failed to predict particularly large axis lengths, as well as very short axis lengths. The model error also yielded slightly higher errors with means of 0.10 m for each hand ($median_{left} = 0.08$, $median_{right} = 0.09$), and paired t-test showed speech input to yield significantly better performance (left hand: $p < .05$, right hand: $p < .001$) .

6.3.2.3 Arm swivel

Arm swivel measures the angle of rotation around the axis between shoulder and wrist. Increasing swivel angle for the left arm (moving the elbow out) means a higher positive value, whereas increasing the right arm's swivel means increasingly negative values. For the left arm, we found a mean angle of 14.55 degrees ($std = 15.79$), and our model yielded a mean error of 11.44 ($median = 8.54$) . For the right arm, the mean swivel angle was -21.27 ($std = 13.29$) and our model yielded a mean error of 9.57 ($median = 6.96$). (see also Figure 6.5 (top)). The mean error reductions with respect to random sampling were 32% (left hand) and 27% (right hand), and median reductions were 36% (left hand) and 22% (right hand) (see Table 6.2).

6.3.2.4 Hand opening

Hand opening averaged 16.74 cm ($std = 4.20$) and 16.89 cm ($std = 4.15$) for the left and right hand, respectively, and the corresponding mean model errors were 1.45 ($median =$

0.96) and 1.38 (*median* = 0.99) (see also Figure 6.5 (bottom)). As noted in Table 6.2, this meant mean error reductions, with respect to random sampling, of 37% and 27% for the left and right hand, respectively, and median reductions of 36% and 22%.

6.3.2.5 Statistical error evaluation

Paired t-tests showed that all error reduction with respect to random sampling were statistically significant (all $p < .001$). Wilcoxon tests revealed that path length prediction errors were lower than for all other parameters except arm swivel ($p < .001$ for all but left acceleration ($p < .05$)). Arm swivel errors were lower compared to all parameters except path length ($p < .001$ for all but left acceleration ($p < .05$)). Initial acceleration as well as hand opening yielded lower errors than velocity (all $p < .001$).

6.3.3 Discussion

In this first part of our work, we sought to examine which gesture parameters may be predicted well from a speech signal and may therefore be well accounted for by a speech-to-gesture generation model. For this, we explored five different gesture parameters.

Firstly, we explored gesture velocity which has been used in previous work on gesture generation from speech [4, 120]. However, interestingly, we found this to be a difficult parameter to model from speech. While there does appear to be an underlying relationship between the speech as represented by the GeMAPS, it proved to be difficult to capture velocities farther from the mean, i.e. we could not capture the full variability of velocities. We explored an additional measure of the gesture kinematics, the acceleration to the first major velocity peak. Acceleration was predicted more accurately than velocity. We found that the model often successfully detects high initial acceleration; common errors are failing to capture high initial acceleration of the left hand (non-dominant hand) and instead only capturing this for the right hand, as well as not modelling very high values. Avoidance of high value predictions can be expected due to the low frequency of these values overall; the model would be penalized strongly for wrongly predicting large values, and rewarded only in the infrequent cases of true high values. Oversampling high values or employing data augmentation to increase frequency of high values may encourage more diverse predictions.

For modelling gesture size, we explored two alternative measures, path length and major axis length. Path length captures the total accumulated distance travelled by the hand during a gesture. We reasoned that gestures with larger lengths take more time to complete, i.e. a gesture stroke with large path length would be associated with a longer speech segment. Therefore, in order to assess how well we can model path length from speech-inherent information, we first ran a baseline prediction model conditioned on only the length of the speech signal. We found that the length of the speech signal is highly correlated with gesture path length. Indeed, comparing this to using GeMAPS speech features as model input, we saw no obvious error change. However, when qualitatively assessing the model predictions, we could see that the model using only speech length failed to predict long path lengths much more frequently (see Figure 6.3), and a statistical test confirmed this improved performance.

Our second measure for gesture size was the major axis length, defined as the distance between the minimum and maximum point of the gesture stroke. We found relatively good prediction results, but some model weakness in predicting large values. Based on our prediction results for path length and major axis length, with our control conditions using only speech length as input, we conclude gesture size may be inferred from the speech signal to some extent.

Our modelling results also emphasize the difficulty of the speech-to-gesture generation problem. Even with a highly-reduced data complexity of just one gesture descriptor rather than many skeleton joints, accurate modelling was difficult. Furthermore, we unexpectedly found that we could model all parameters similarly well. Performance for arm swivel and hand opening was on par or better than performance for gesture kinematics (velocity and acceleration). Arm swivel angle prediction accuracy was surpassed only by path length.

Next, we will assess the perceptual importance of these gesture parameters for producing speech-gesture coherence.

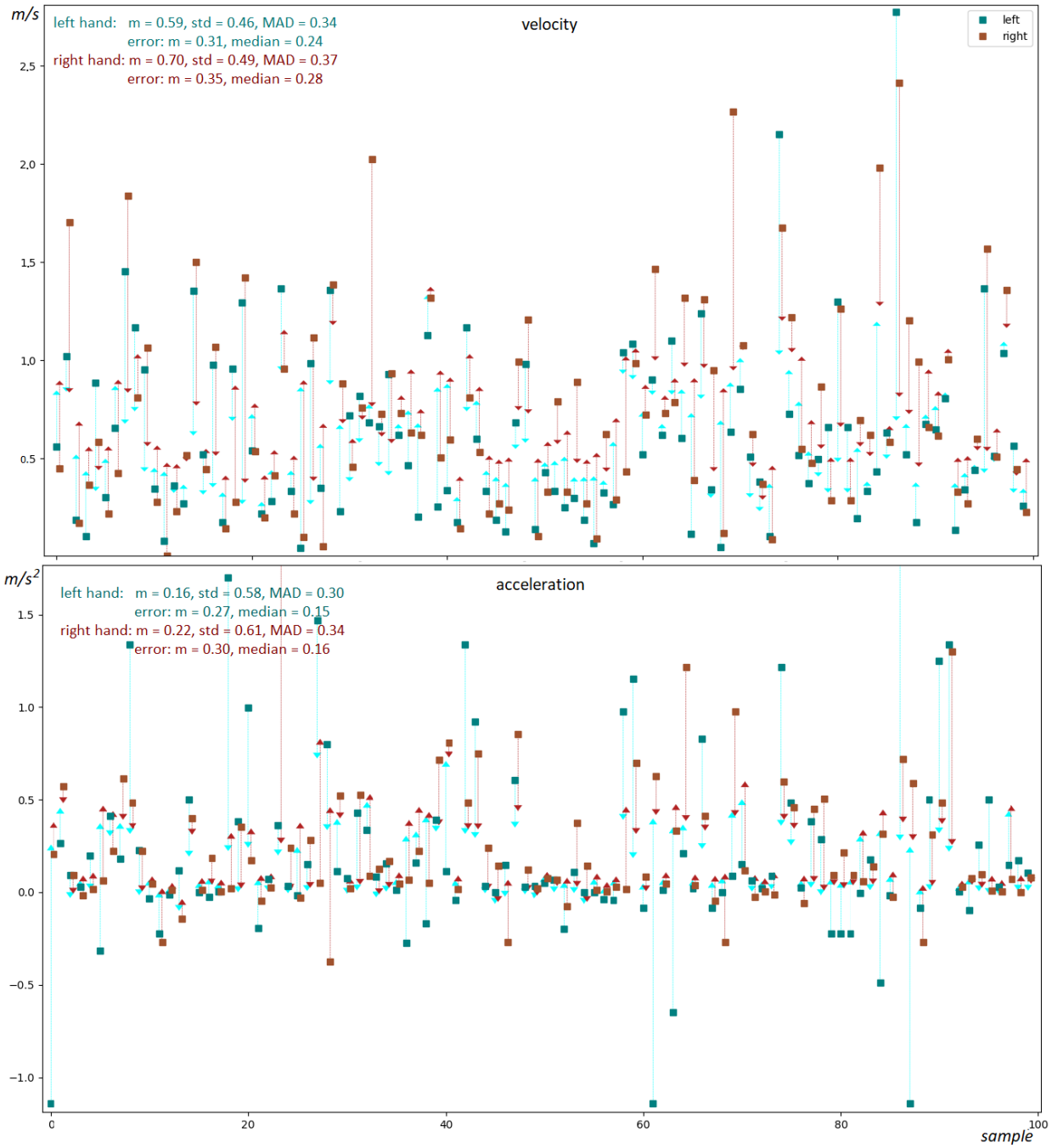


FIGURE 6.2: Gesture kinematics. Arrows indicate the size and direction of the prediction error. Top: Maximum velocity, bottom: initial acceleration. Plotted are the true values for 100 gesture stroke samples and lines indicate the respective prediction error. It is visible that very high true values yielded the largest prediction errors; the model avoided prediction of extreme values. The model generally predicted closely correlating values for the right and left hand, with slightly higher values for the right.

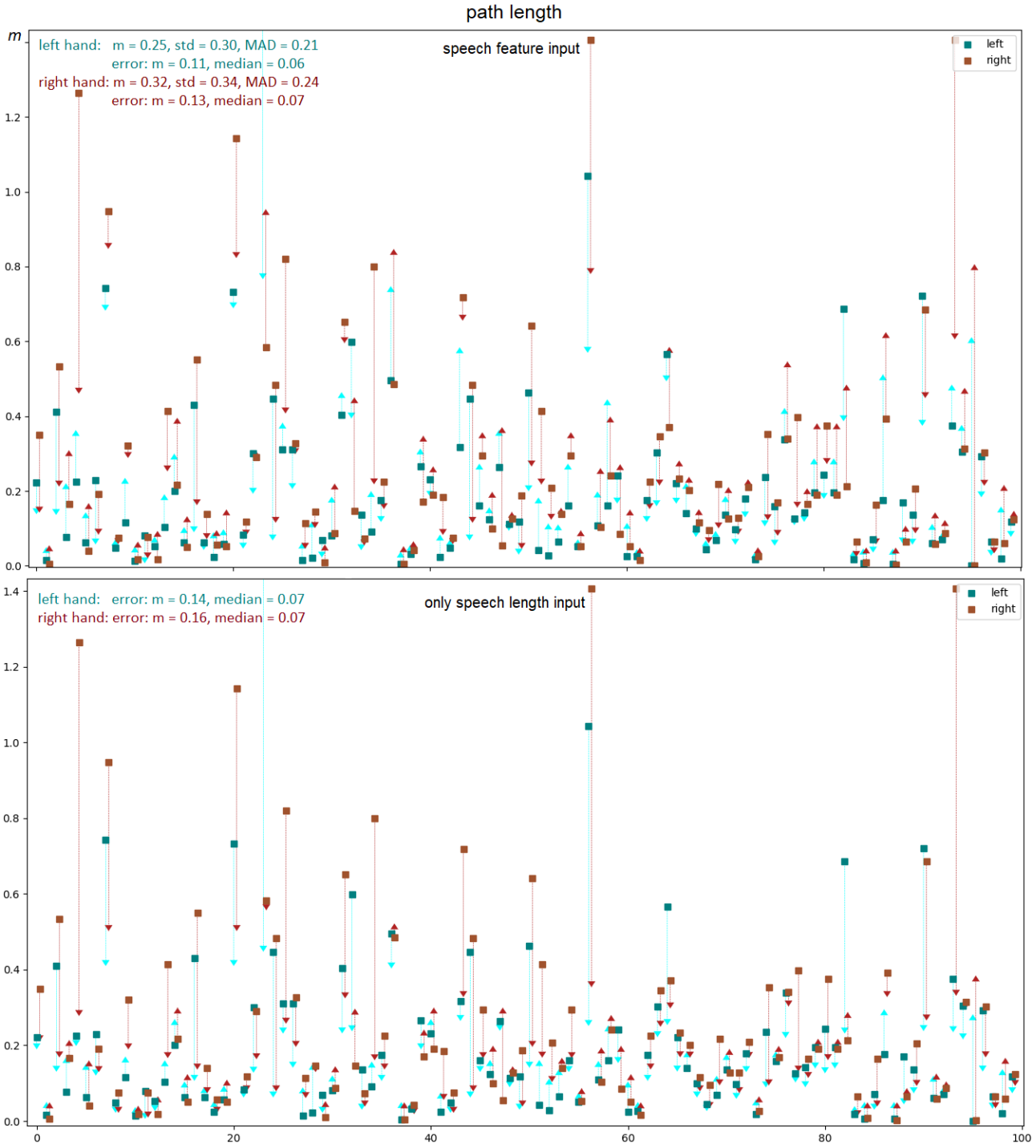


FIGURE 6.3: Path length prediction errors. Top: Predictions for model using GeMAPS speech features. Bottom: Prediction for model only using speech length encoding. Using only speech length fails to predict longer path lengths.

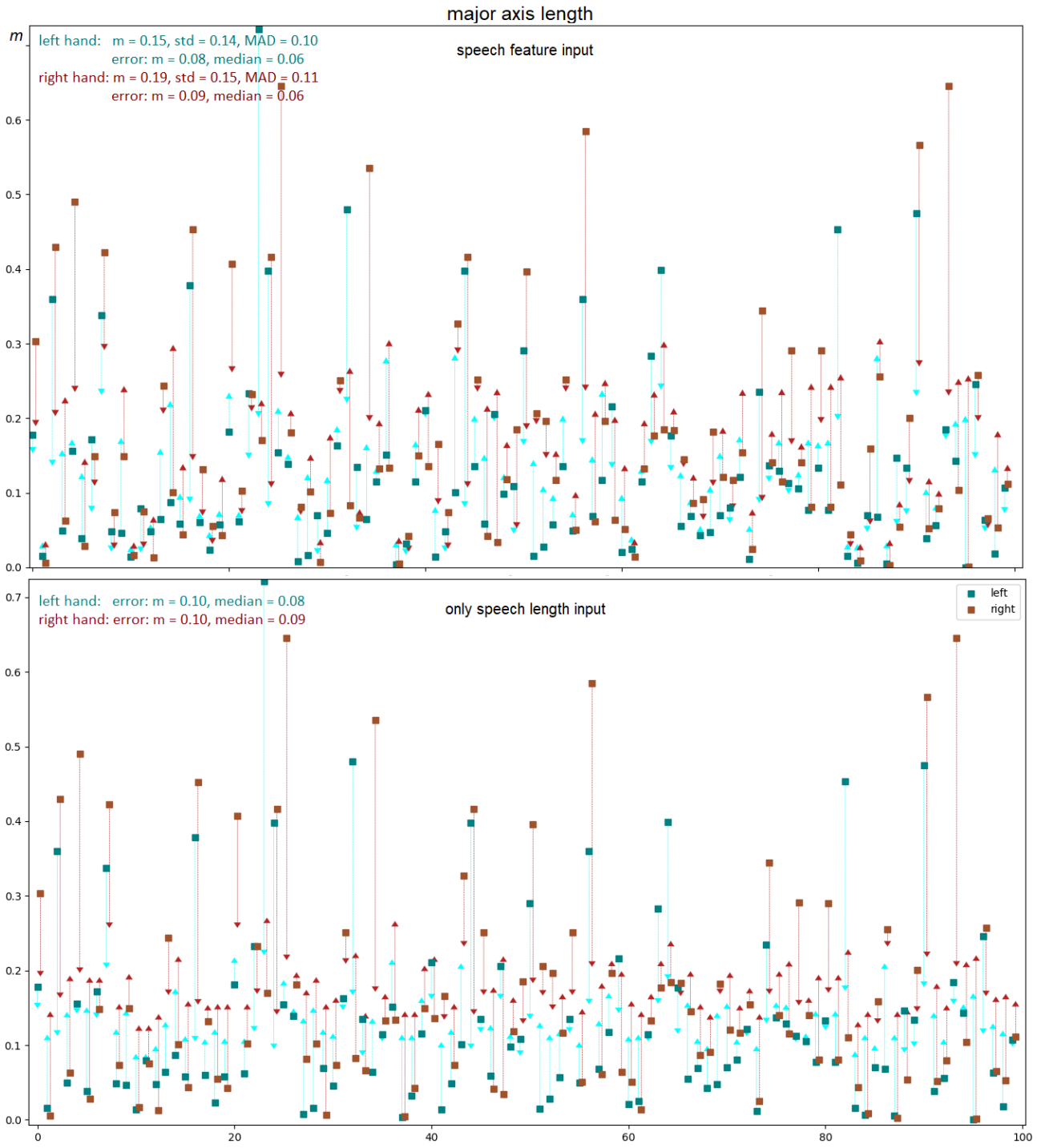


FIGURE 6.4: Major axis length prediction errors. Top: Predictions for model using GeMAPS speech features. Bottom: Prediction for model only using speech length encoding. Using only speech length yields axis length predictions closer around the average values.

FIGURE 6.5: Arm swivel and hand opening. For hand opening, we can see an apparent division into two classes; this is due to speaker differences of the two datasets we trained on, with the speaker of dataset 1 representing the higher values. However, the model was trained on data normalized within each dataset, hence this division is only visible due to output-denorming for plotting.

6.4 Gesture parameter evaluation

We designed an empirical evaluation of the impact of the gesture parameters on perception. We assessed people's judgment of the gesture expression regarding its suitability for the expressed speech. This is important for gesture synthesis, telling us which parameters are most important in achieving a natural-looking gesture expression. We tested the perceptual impact of our gesture parameters by creating variations that increase or decrease them, as described below.

6.4.1 Stimuli creation

Artificial stimuli were created through a three step process. First, the variation in the source data was measured. Second, clips were selected that best represent high and low variations within this. Third, these clips were algorithmically modified to fully match the desired high and low performance.

The process began by computing the natural variation of each of our parameters within the gesture database. For this, we calculated the 25th percentile marker as a lower bound, and the 75th percentile marker as the upper bound. Samples below the lower bound were defined as having a *low* expression, and samples above the upper bound were defined as having a *high* expression of a given parameter. The bounds of this parameter segmentation are visualized in the data distribution plots in Figure 6.6.

In a second step, we randomly selected short gesture sequences of about 10 seconds, similar to previous work in expressive motion perception [81, 93]. A 10-second timeframe has been reported to be sufficient for participants to make judgements about conversing agents [9]. For *low* sequences, we chose sequences that contain *low* parameter expressions. However, as there are practically no 10 second sections in the database of only *low* expression, we allowed the sequences to contain some *medium* expression (values below the upper bound), but gave preference to gesture sequences with the highest percentage of *low* samples. Equivalently, for *high* sequences, we chose sequences containing mainly *high* expression, allowing some *medium* expression samples. This biased selection ensured that the edited clips were as different as possible from the source clips (i.e. error maximizing).

In the third step, we created the parameter manipulations. For *low* sequences, we increased the parameter expression to *high* but keeping within the found natural limits. For *high* sequences, we decreased the parameter expression to *low*. We selected 5 samples each for the *low* and the *high* manipulations of each parameter. As baseline samples, we randomly selected 10 sequences that remained un-manipulated.

All samples were generated with custom animation software that uses a motion parametrization similar to Neff and Kim [162] and IK tools to generate variations of the input motion capture data. The software takes as input the motion data and the corresponding stroke labels and synthesizes preparation (bringing the hands into position for the gesture) and retraction (returning the hands to a rest position) phases for the strokes, proportionally matching the stroke speed. By using this software to determine preparations and retractions, we avoided problems of e.g. two lengthened gestures not leaving time for an originally present retraction between them. If a manipulation was applied, it was applied to the stroke phase.

We restricted our data selection to the hand-annotated sections of dataset 2. Including dataset 1 in this step would have require manually correcting automatically determined stroke labels to ensure correct boundaries and a controlled study.

All stimuli can be viewed at
`https://youtube.com/playlist?list=PLLrShDUC_FZzhemzr0g1ekt1jz45-y_u3`.

6.4.2 Experiment

The experiment was designed with the Unity3D game engine and the Virtual Human Toolkit (VHTK) [167]. The displayed character was Brad from the VHTK, (see Figure 6.7), producing regular eye blinks, lip synchronisation, as well as an idle motion for the body excluding the arms and hands. We chose this character for its low enough level of realism to match somewhat synthetic motion while still being a reasonable choice for real world applications. In each experiment trial, participants first watched a ∼10 second clip of the character acting out one of the gesture sequences. Following the clip, participants were asked the following question:

"How well did the expressive quality of the gestures match the expressive quality of the speech?"

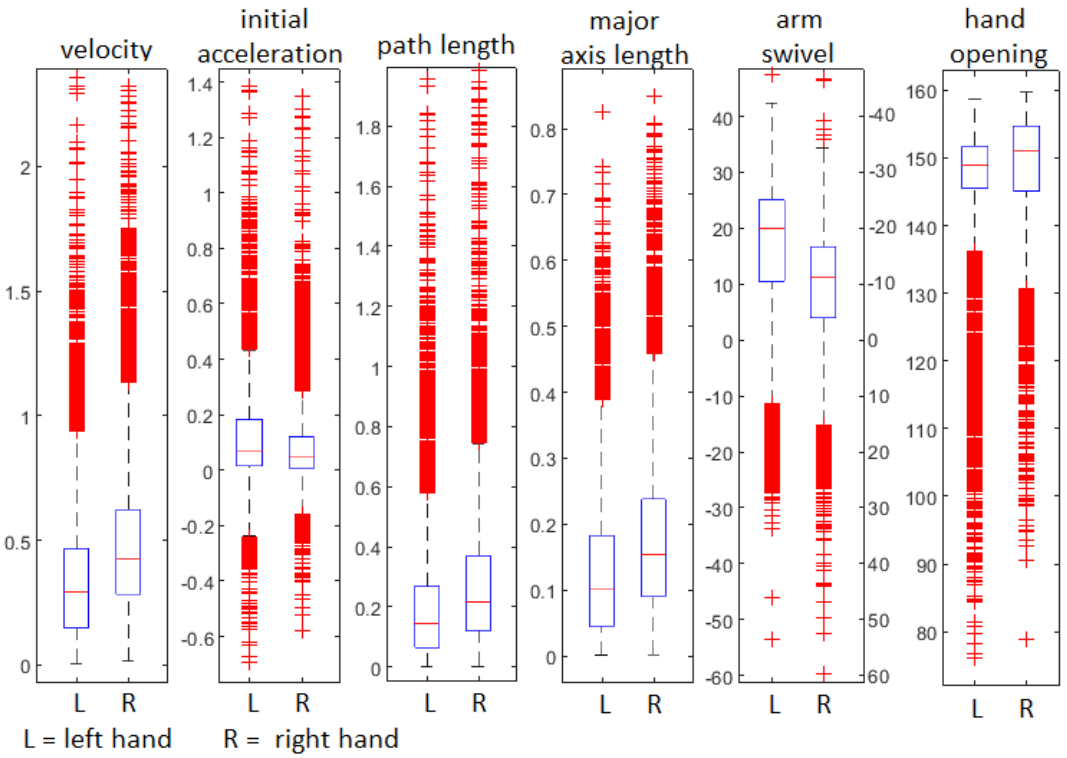


FIGURE 6.6: Distribution of the gesture parameter values for dataset 2 (3.2). The boxplots indicate the 25th and 75th percentile, the bounds we defined for *low* and *high* parameter expression. Red markings indicate outlier values.

This question was specifically designed to motivate participants to focus on the expression of the gestures; we did not want participants to judge the semantic entropy of the gesture sequence. A 7-point Likert scale was provided as a rating scheme. Participants first completed 5 example trials for which responses were not recorded. This was in order to establish an expectation of the gesture quality variation in the experiment and to familiarize the participants with the rating scale. Following the example trials, participants completed 60 experiment trials (5 samples for each of the 5 parameters, with 2 expression manipulations each, plus 10 baseline samples), presented in random order.

The online experiment was distributed via university mailing lists and social media. We collected data from 60 participants (23 females, 36 males, 1 other gender, ages 18-59 years, $M = 26.4$, $SD = 9.1$), all of whom gave informed consent regarding their participation. All participants reported sufficient English proficiency (35 "native", 20 "fluent", 3 "very good", 2 "good").

FIGURE 6.7: Visualization for the perceptual experiment. On the left is the recorded actor from database A; on the right is the character used in the perceptual experiment.

6.4.3 Results

The study consisted of two factors, the *parameter* that was modified and the *direction* of the modification. The first factor had 11 conditions, with mean ratings summarized in Figure 6.8, and the rating score distribution further explored in Figure 6.9. The second factor had two levels, increase and decrease. There was a main effect of modification parameter ($p < .001$) and a significant interaction of modification parameter and modification direction (increase versus decrease) ($p < .001$).

We analyzed the data further by treating the rating scores as ordinal data and fitting a cumulative link model, using clm from the R ordinal package [168]. All modification conditions were rated significantly lower than the no modification condition (all $p < 0.001$). Decreasing gesture size was rated significantly worse than increasing gesture size ($p < .001$). Decreasing hand opening was preferred over increasing ($p < .001$). Increasing hand opening received the lowest rating compared to all other conditions ($p < .01$ for velocity manipulations and gesture size decrease, $p < .001$ for all others).

The complete results are detailed in Table 6.3.

6.4.4 Discussion

We found that all our gesture modification had a significant perceptual effect. Unmodified gestures were preferred over all modification conditions, indicating some perceptual relevance for each of the five gesture parameters.

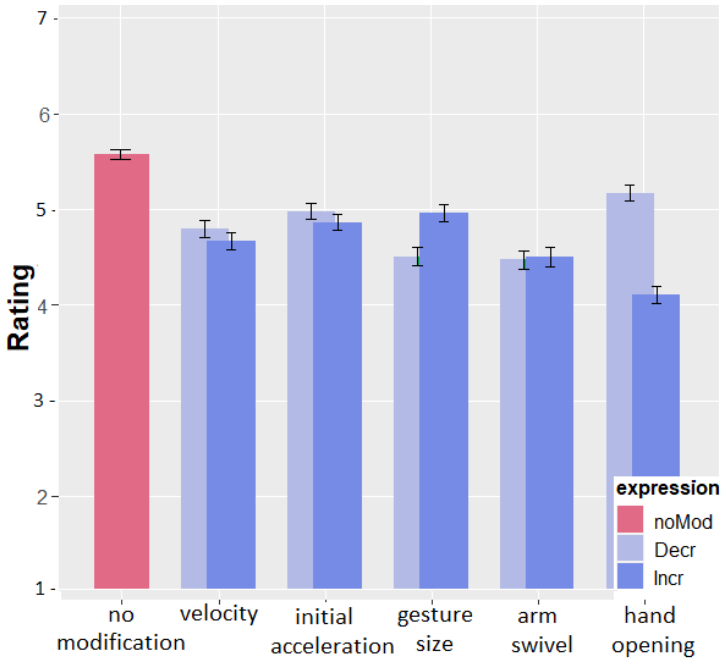


FIGURE 6.8: Mean rating scores for all experimental manipulations. Unmodified gestures received the highest average rating, and increased hand opening the lowest.

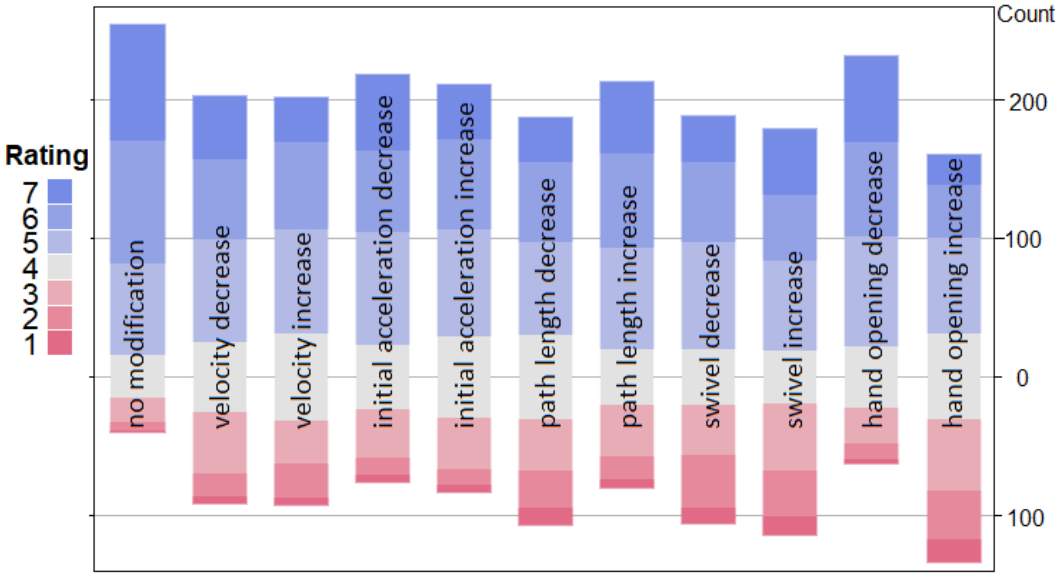


FIGURE 6.9: Stacked bar chart of all given ratings. The no-modification condition is scaled by 50%. Plotted is the frequency of responses for the 7 rating scores. (The y-axis represents the frequency of responses).

Altered gesture kinematics, as described by gesture velocity and initial acceleration, significantly worsened speech-gesture match, with the slowing-down modification yielding similar ratings as the sped-up modification.

For modified gestures, we found that enlarged gesture size was preferred over reduced size gestures. Enlarged gesture size was further preferred over a number of other

TABLE 6.3: All results for the perceptual experiment. Indicated are both significant and non-significant condition differences (plotted in Figure 6.8 and Figure 6.9). + means the row condition was rated higher, - means lower rating. n.s.=not significant, marg.=marginal significance of $p = .06$.

	no mod.	vel.↓	vel.↑	init.acc.↓	init.acc.↑	size↓	size↑	swivel↓	swivel↑	hand↓	hand↑
no mod.	-	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$	$***^{+}$
velocity ↓	$***^{-}$	-	n.s.	n.s.	n.s.	$*^{+}$	n.s.	$*^{+}$	marg.	$**^{-}$	$***^{+}$
velocity ↑	$***^{-}$	n.s.	-	$*^{-}$	n.s.	n.s	$*^{-}$	n.s.	n.s.	$***^{-}$	$***^{+}$
init. acc. ↓	$***^{-}$	n.s.	$*^{+}$	-	n.s.	$***^{+}$	n.s.	$***^{+}$	$***^{+}$	n.s.	$***^{+}$
init. acc. ↑	$***^{-}$	n.s.	n.s.	n.s.	-	$*^{+}$	n.s.	$**^{-}$	$*^{-}$	$**^{-}$	$***^{+}$
size ↓	$***^{-}$	$*^{-}$	n.s.	$***^{-}$	$*^{-}$	-	$***^{-}$	n.s.	n.s.	$***^{-}$	$**^{+}$
size ↑	$***^{-}$	n.s.	$*^{+}$	n.s.	n.s.	$***^{+}$	-	$***^{+}$	$***^{+}$	n.s.	$***^{+}$
swivel ↓	$***^{-}$	$*^{-}$	n.s.	$***^{-}$	$**^{-}$	n.s.	$***^{-}$	-	n.s.	$***^{-}$	$**^{+}$
swivel ↑	$***^{-}$	marg.	n.s.	$***^{-}$	$*^{-}$	n.s.	$***^{-}$	n.s.	-	$***^{-}$	$**^{+}$
hand open ↓	$***^{-}$	$**^{+}$	$***^{+}$	n.s.	$**^{+}$	$***^{+}$	n.s.	$***^{+}$	$***^{+}$	-	$***^{+}$
hand open ↑	$***^{-}$	$***^{-}$	$***^{-}$	$***^{-}$	$***^{-}$	$**^{-}$	$***^{-}$	$**^{-}$	$**^{-}$	$***^{-}$	-

modifications, while reduced gesture size showed the opposite trend. Machine learning models for gesture generation are often trained with a mean-squared error loss [4, 120, 169], commonly leading to smaller than natural gestures due to convergence to the mean pose. Our perceptual results give further motivation to move away from such traditional model training approaches. Recent works have proposed alternative approaches [124, 127, 160].

There was a large effect of hand opening, with the open, flat hand rated significantly lower than all other modifications. Gesture sequences with decreased hand opening were preferred over most other modifications. In line with the strong effect we found for manipulating hand shape, Wang et al. [83] have reported large effects of hand pose on personality perceptions. Modelling finger motion is a complex problem due to the high dimensionality of the hand skeleton; when accurate hand shape prediction is not possible, based on our results, we suggest animating slightly flexed fingers rather than straightened fingers.

Modifying arm swivel angle in either direction elicited relatively low preference ratings, indicating this to be an important factor in believable gesture synthesis.

6.5 General Discussion

In this chapter, we investigated the relationship between speech and gesture expressivity. Gesture generation approaches often assume some underlying connection between

modalities by training black-box type models, feeding speech data and outputting high-dimensional and complex skeleton motion data. Due to the limited success, we here aimed to assess in more detail how speech relates to gesture motion. Based on a literature review, we first determined a number of parameters to describe a gesture. We then assessed the speech-gesture parameter relationship in two ways.

First, we used machine learning, specifically recurrent neural networks, to phrase the question as a prediction problem of speech to gesture. We trained separate models for each gesture parameter, each working solely on the audio speech signal as input. By judging the successes or failures of the model predictions, we gained a measure of how well the speech signal relates to a given gesture parameter. Results indicated that all gesture parameters were predicted above chance, but there was variance in how well they were predicted. For example, size and initial acceleration of a gesture were predicted better than its velocity. Arm swivel predictions, surprisingly, surpassed all other measures but path length. Our results also indicate the remaining difficulty in modelling the speech to gesture relation. Previous work on gesture generation has reported good adherence of their model to the acceleration distribution of a dataset [119, 120], however, our results indicate that the correct acceleration and the correct time matters, and generated gestures should hence be assessed in a gesture-specific rather than output-general manner.

Second, we conducted a perceptual study to assess the relevance of each gesture parameter for gesture synthesis. For this, we manipulated the expression of each parameter and tested the impact on the perceived match of speech and gesture. Observers were sensitive to all variations in parameters away from the original performance, indicating that each of our chosen parameters is important in realistic gesture synthesis. Hand pose showed to be particularly important, with flat, open hands being viewed especially negatively, and more flexed fingers being preferred. Regarding gesture size, we found enlarged gestures being preferred over reduced gestures.

For gesture parameter prediction, we saw an expected preference of the models to keep predictions somewhat around the mean for all parameter values, infrequently predicting extreme values. Based on our perceptual results, speech-to-gesture training data could be augmented for better results: for example, due to participants' preference for enlarged versus reduced size gestures, and the common problem of reduced-size

gesture output in machine learning models, we could increase the frequency of large gestures within the training dataset. This could be done in three ways: by oversampling large gestures selectively, by oversampling and augmenting large gestures by applying perceptually less salient modifications (e.g. slight acceleration warps), or by applying data augmentation of smaller gestures (artificially enlarging). Additionally, rather than tackling high-dimensional finger motion modelling, simply using slightly flexed fingers is a perceptually reasonable choice.

With this work, we hope to provide better insights into which aspects of gesture may be modelled from speech. We suggest a step toward better evaluation of gesture generation models by providing numeric gesture descriptors that impact the perceived match of the generated gesture, as shown by our perceptual study.

We have previously discussed differences in phase expression (see Chapter 5) and in this chapter have provided an initial assessment of how gesture performance varies between speakers by comparing stroke parameter values between speaker 1 and 2. It would be interesting to include a larger variety of speakers and speaker style. We propose the use of the here used gesture parameters to investigate how gesture expression can, or cannot, be compared across speakers.

While this work focused on performance variation, it is also important to correctly match the semantics of the gesture with the spoken text. Systems that generate gesture from speech signals will ultimately need to match both style and content.

In the following chapter, we will draw upon the insights gained here on how gesture expression relates to the concurrent speech signal to design a novel gesture generation system. By using the gesture parametrization presented here, we can avoid the problem of training a model on explicit, high-dimensional skeleton data while ensuring that the produced gestures adhere to perceptually relevant constraints of the speech-gesture relationship.

Chapter 7

Gesture Matching System

In this chapter, we present a gesture generation system based on the expressive parametrization of gesture motion proposed in the previous chapter.[1] We evaluated our system in three perceptual studies, comparing our output to a number of baseline models as well as to state-of-the-art machine learning models.

7.1 Introduction

We propose a novel speech-to-gesture system based on matching expressive gesture parameters to prosodic speech information. Instead of modelling an implicit relationship between high–dimensional, exact joint poses and the speech signal, we utilize a higher level representation of the gesture motion through expressive parameters that were shown in the previous chapter to be associated with the speech signal as well as being perceptually important to the quality of the speech-gesture match. These gesture parameters include gesture velocity, acceleration, size, arm swivel angle and extent of hand opening.

We present a merged approach of machine learning and database sampling to produce realistic gesture form. Specifically the machine learned models from the previous

[1]The contents of this chapter are being published at the International Conference on Autonomous Agents and Multiagent Systems 2021 (AAMAS'20) (Listed Publication 5) and is currently under review for the International Conference on Computer Animation and Social Agents (CASA'21) (Listed Under Review 1). The second author, Michael Neff, contributed to the study design and provided the animation engine.

chapter are used offline to establish the speech-gesture relationship by encoding the relationship between acoustic speech features and expressive gesture parameters. Online, given a new speech input, the models estimate gesture parameters that are then used to search a large gesture database for the gesture best matching the predicted expressive parameters. We built the database of gestures by segmenting our datasets from Chapter 3 into individual gestures, resulting in a set of over 23,000 individual high-quality motion-captured gestures.

We evaluated our method with three perceptual experiments. In the first experiment (Section 7.3), we assessed the appropriateness of our gesture selection method by comparing it to two two baseline conditions. The first baseline used the same gesture timings but selects gestures without considering expressive gesture parameters. This allowed us to assess the validity of gesture selection method with respect to speech match. The second baseline disregarded both parameter match as well as speech-coherent gesture timing. This let us assess the relative importance of gesture timing. In the second experiment (Section 7.4.2), we combined our method with predicting gesture timing from speech and compare our results to a baseline condition of random gesture selection. In the third experiment (Section 7.4.3), we compared our method to five current generative machine learning models, namely the entries to the GENEA gesture generation challenge [132].

7.2 Synthesizing a gesture sequence

For synthesizing new gesture behavior, we took as input an arbitrary-length speech segment with associated desired gesture timings. We defined the desired gesture timings as the stroke timings of the associated motion data, as estimated by the stroke classifier, but we will provide a speech-based method in Section 7.4. By using the true stroke timings for a gesture, we could compare our method of gesture selection directly to the ground truth gesture sequence; using different gesture timings would conflate perceptual effects of timing and gesture form.

The gesture timing provides a sequence of empty motion slots with associated speech data, each to be filled with a gesture stroke from the database (see Figure 7.1). The first step in determining the gesture choice is computing a set of desired gesture parameters.

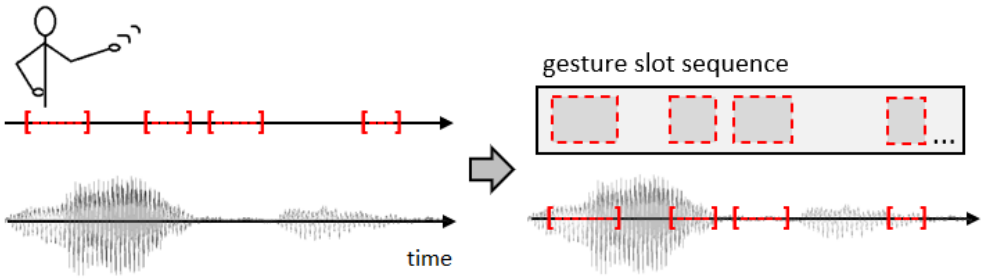


FIGURE 7.1: Gesture timing from motion segmentation. Within the continuous motion signal, the original gesture stroke timings were determined with the method of Chapter 5. The resulting timings provide a sequence of empty motion slots, each to be filled with a gesture stroke from the database. Each gesture slot is associated with an original speech segment.

Theoretically, we can use any parameter automatically computable for a motion segment, but chose to use the five gesture parameters shown in Chapter 6 to be associated with the speech signal as well as significantly impacting perceptions of the quality of the speech-gesture match. As before, these five parameters are (1) the gesture velocity, (2) the size of the initial acceleration peak, (3) the gesture size measures by the total completed path length, (4) the arm swivel (the rotation around the axis between shoulder and the wrist, bringing the elbow closer or further from the body), and (5) hand opening, describing how open or closed the hand shape is (calculated as the mean distance of the finger tips from the base of the wrist). For the computation of these parameters from the speech signal, the GeMAPS prosodic features [161] were computed automatically using openSMILE [151]. The process of parameter prediction from prosodic feature is described in detail in Chapter 6. The distribution of the five parameters for both datasets is visualized in Figure 7.7.

Given this set of desired gesture parameters for a speech segment, we searched the database for the best match. First, each gesture in the database was assigned a rank number with respect to each of the five parameters; e.g. the gesture with the closest-matching velocity would receive velocity rank 1, and the gesture sample with the worst-matching velocity received velocity rank 23,700. Each gesture hence had 5 rank values, one for each parameter. For selecting a gesture, each rank value was weighted to decide the importance of a parameter, before combining all 5 rank values into a total match rank. We determined the parameter weights based on how well a parameter can be predicted from speech and its perceptual importance for speech-gesture match. For example, gesture size was predicted best from the speech signal, followed

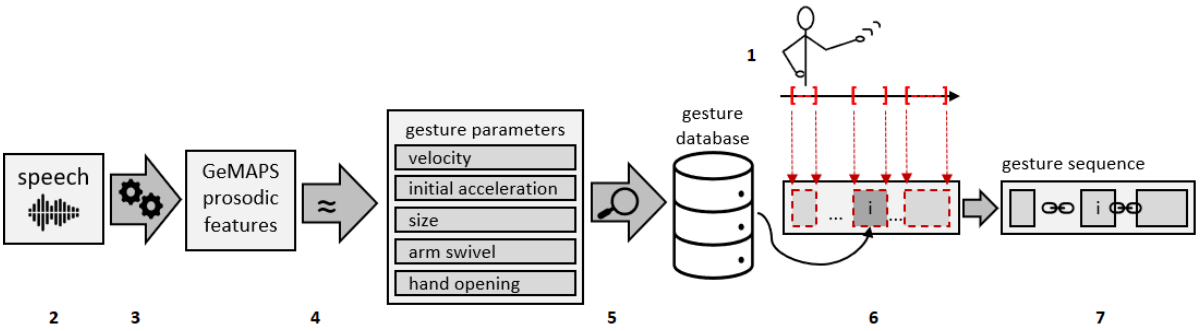


FIGURE 7.2: Overview of our gesture generation system. (1) Gesture timings are determined through motion segmentation (later replaced by speech analysis in Section 7.4). (2) The system receives as input the speech audio for a gesture segment. (3) Prosodic speech features in the form of GeMAPS are extracted automatically. (4) From the prosodic features, the desired values for the five gesture parameters are estimated. (5) The database is searched for the gesture with the closest matching parameter values. (6) The best matching gesture is inserted at the desired gesture position. (7) Synthetic preparation and retraction phases are generated to link the gestures in the sequence.

by arm swivel, and acceleration was predicted better than velocity; hand shape had a strong perceptual impact on the perceived gesture match (see results Chapter 6). We defined the following parameter weights: $weight_{velocity} = 0.6$, $weight_{acceleration} = 0.8$, $weight_{size} = 2$, $weight_{swivel} = 1.5$, $weight_{hand} = 1$.

The complete gesture synthesis process is visualized in Figure 7.10 and the gesture matching algorithm is detailed below in pseudo-code:

Offline, before first use:

for each gesture g in gesture_database:

for each parameter p in gesture_parameters:

calculate p(g)

Online, to synthesize a new sequence:

for each gesture slot s in gesture_sequence:

return max(rowsum(W*(P(s)-P(G))))

Where P(G) denotes the array containing all parameter values of all gestures G in the database, W is the array of the 5 parameter weights, and P(s) are the estimated parameters of a speech segment s. All parameter values are normalized to the range of 0-1. The online computation component has a time complexity of $O(s*(3np+n))$, with s as the number of gesture slots in an input speech segment, n the number of gestures in the database, and p the number of gesture parameters.

In order to improve the smoothness of gesture transitions, we returned the top 10 gestures in the match rank algorithm and then chose a gesture that allowed a reasonable transition. This was done by calculating the distances of the top 10 gestures' starting wrist positions from the previous gesture's end positions and, taking into account the available time for transition, selecting a gesture that allows for a realistic transition speed.

As we used pre-determined stroke durations from the labelled stroke phase input, we also constrained gesture selection to gestures with similar duration, resulting in an average number of about 1200 gesture samples to search for the best match rank. Without this constraint, a selected gesture could overlap with the next stroke slot, resulting in different gesture timings than the ground truth sequence we wanted to compare to.

7.3 Experiment I - Gesture selection validation

In order to evaluate the success of our method in creating gesture sequences that match the speech expression, we performed a perceptual experiment comparing our method in an ablation manner to two baseline methods, as well as to the ground truth gestures.

7.3.1 Experiment conditions

Our first baseline comparison used the same stroke timing but selected gestures agnostic to the desired parameter values, i.e. the first baseline method (unmatched) was equivalent to our method without calculating the match rank, the pseudocode noted in Section 7.2.

The second baseline method did not use the same stroke timing (unmatched & untimed); it scrambled the order of all the timings within our test dataset, resulting in the same realistic stroke and between-stroke durations, without preserving the speech-gesture synchrony. Specifically, to reorder the timings for baseline 2, we took the sequence of ground truth stroke timings, containing both duration and amount of time to the next stroke, and randomize their order. This ensured realistic gesture timings while

breaking the speech-gesture synchrony. As in the first baseline condition, these stroke slots were filled without calculating the rank match.

Finally, the ground truth condition selected the true stroke for each gesture slot. This therefore reflected the true gesture behavior, but using synthetic preparation and retraction. All conditions are visualized in Figure 7.3.

For all methods, the selected gestures were combined into coherent sequences using animation software based on the open-source animation environment DANCE [163], taking as input motion data and corresponding stroke labels and synthesizing preparation and retraction phases for the strokes by using splines. The preparation phase brings the hands into position for the stroke phase, and the retraction returns the hands to a rest position. Preparation and retraction proportionally matched the stroke speed. If there was not enough time for a retraction before the beginning of the next gesture, only a transitional preparation was synthesized instead.

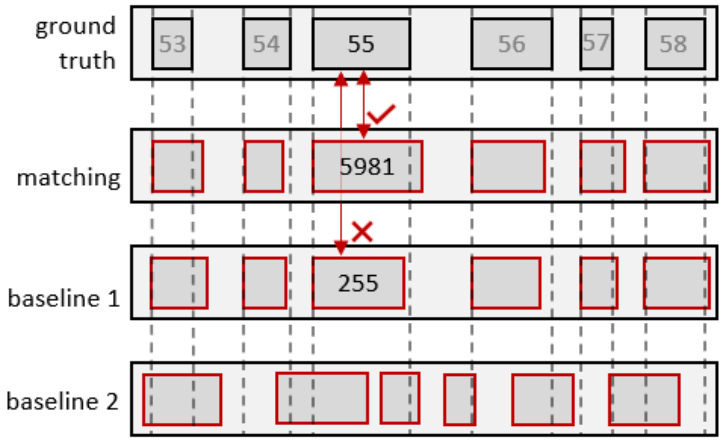


FIGURE 7.3: Experiment conditions. We created gesture sequences with four different methods. Dashed lines indicate the ground truth stroke boundaries. The ground truth gesture sequence produced the ground truth stroke at the ground truth time. The matching condition searched among strokes with similar duration and found the candidate with the best predicted parameter match. For example, to fill stroke slot 55, we used the co-occurring speech to compute the 5 gesture parameters; stroke #5981 was detected to have the best matching parameters among all strokes with similar duration and is hence inserted into the sequence. For baseline 1, we searched for a stroke with similar duration and insert it at the correct time. Baseline 1 did not respect desired gesture parameters, stroke #255 followingly does not match the predicted parameters. Baseline 2, did not respect the ground truth gesture timing and picked any strokes from the database to populate the gesture sequence.

FIGURE 7.4: Example of a generated gesture sequence. After performing the first gesture (1), the arms and hands are moved to a predefined rest position through spline interpolation (2). When there is not enough time to move through the rest pose, interpolation instead just creates a trajectory from a stroke end position (3) to the next stroke's start position (4).

7.3.2 Perceptual study

We distributed an online experiment was via university mailing lists with an incentive of a 50 Euro raffle voucher. A total of 109 participants completed the experiment, 54 in the experimental and 55 in the control group. (33 females, 74 males, 2 other, ages - years, $M = 21.4$, $SD = 5.6$), all of whom gave informed consent regarding their participation.

For creating our experiment stimuli, we animated gesture sequences on the Brad character from the open-source Virtual Human Toolkit (VHTK) [167] (see Figure 7.4), using the Unity3D game engine (as in the perceptual study of the previous Chapter 6). In addition to the arm gesture motion, the character displayed some idle lower body motion, head movements, regular eye blinks, and lip synchronisation. In each trial, the participant first watched a video clip of about 10 seconds, a time frame previously shown to be sufficient for making judgements about conversing agents [9]. The participant then answered the same question as used in the previous Chapter 6 on a 7-point Likert scale: *"How well did the expressive quality of the gestures match the expressive quality of the speech?"*.

Participants first completed 4 example trials, followed by 48 experiment trials (6 clips for each of the 2 speakers, for 4 conditions), presented in random order. Each clip contained a different speech segment.

For assessing if between-condition differences are caused by the variance in gesture timing or speech rather than gesture form, we ran a separate control condition that used

the same 48 speech segments, but always presented the ground truth gesture animation. With this, we sought to ensure that the variation in gesture timing, which is used in synthesis, and the spoken utterance alone was not the driving factor behind variation in participants' ratings of clips, but rather the relation of speech utterance and gesture motion. The control condition was completed by a separate set of participants.

All stimuli can be viewed at
`https://youtube.com/playlist?list=PL04OYHqbFqt07NKOl3Qua81JJcJ0WNttZ`

7.3.3 Results

Statistical analysis of the results of the perceptual experiment was performed by treating the Likert rating scores as ordinal data and fitting a cumulative link model, using clm from the R ordinal package [168].

An ANOVA of the estimated model revealed a main effect of condition ($p < .001$), with an effect size measured by Wald Chi Square $\chi^2 = 190.1$. The ground truth condition was rated significantly higher than all other conditions (all $p < .001$), indicating that the ground truth gestures were preferred over any alternative, as expected (rating score $mean = 5.35$). The gesture matching condition (our method) was rated significantly higher ($mean = 4.67$) than both baseline methods (both $p < .001$, baseline 1 $mean = 4.32$, baseline 2 $mean = 4.41$), indicating that matching gestures to speech-predicted parameters indeed increases perceived appropriateness of the selected gesture, in line with our hypothesis. The two baseline conditions were not rated significantly different from each other, suggesting, somewhat surprisingly, that correct gesture timing alone did not improve perceived speech-gesture match.

An ANOVA of the model for the control condition yielded a main effect of condition ($p < .01$) with an effect size of $\chi^2 = 11.74$. In the control group, the clips associated with the ground truth condition (rating score $mean = 4.95$) were not rated significantly different from the clips of the matching condition ($mean = 4.86$) or baseline condition 1 ($mean = 4.82$), and there was no significant difference between the clips of the matching condition or baseline condition 1. Interpreting the lower ratings for the clips used in baseline condition 2 ($mean = 4.66$) is not straightforward as not only gesture selection but also gesture timing differed between experimental and control group. We sought to

establish that the natural variation in the performed gesture timing of the original input clips was not a potential source of the observed variation in our main experiment, and this appeared to be the case for the three conditions that used this information (ground truth, matching and baseline 1). Results are visualized in Figures 7.5 and 7.6.

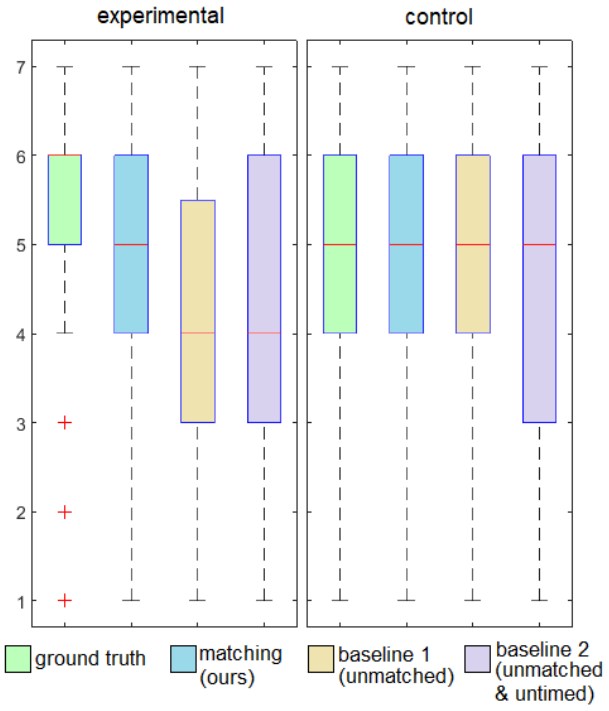


FIGURE 7.5: Boxplots for both experiment conditions visualizing the distribution of rating responses. Red lines indicate the median (most common) rating for each condition, box boundaries illustrate 25th and 75th percentiles, and red crosses indicate outlier responses.

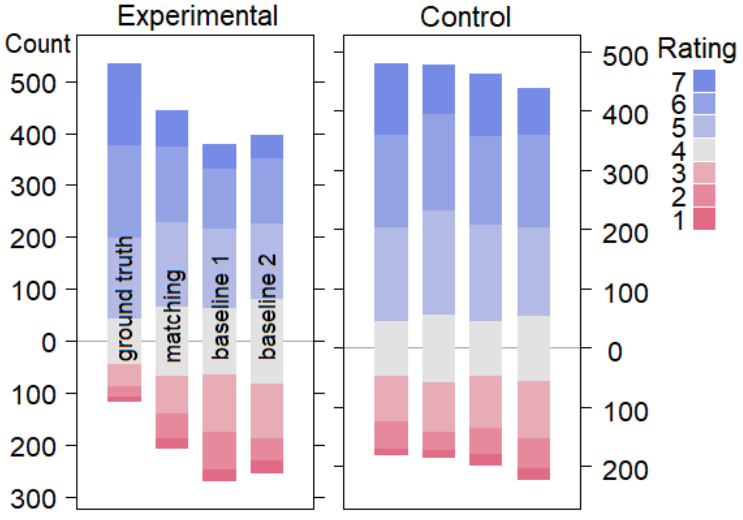


FIGURE 7.6: Stacked bar chart of perceptual ratings for experiment I. Plotted is the frequency of responses for each of the 7 rating scores. (The y-axis is the frequency of responses)

7.3.4 Discussion

Here, we proposed a gesture selection method based on expressive parameter matching. We evaluated our gesture selection method with a perceptual study comparing it to two baseline conditions in an ablation manner, as well as to the ground truth gestures. The first baseline condition selected gestures with the same timing but without respecting desired expressive gesture parameters. By comparing our method to this first baseline, we could evaluate the perceptual validity of our gesture selection method. The second baseline condition disregarded timing synchrony with speech as well as the desired gesture parameters. By assessing the performance of this second baseline, we could evaluate the relative importance of correct gesture timing.

Our results show that our gesture selection method of matching expressive gesture parameters to speech did indeed perform better than the baselines which disregard parameter match, indicating that matching the chosen gesture parameters to the speech significantly improves perceived speech-gesture match. This confirms the proposition of Chapter 6 of the importance of matching expressive gesture parameters to co-occurring speech, as well as asserting the validity of the here proposed avenue for gesture generation.

Interestingly, our second baseline condition, which did not use any timing or speech prosody information, still received relatively high ratings and did not differ significantly from baseline 1, which used correct timing. One potential reason for this is that the speakers in the used datasets produce continuous speech without any significant periods of silence, therefore even untimed gesture is almost always accompanied by speech. In cases without continuous speech, untimed gestures may stand out more negatively. Another potential reason for the insignificant effect of timing in this study could be due to realistic gesture form being enough for reasonably well-liked gesture performance.

A number of participants in our perceptual experiment noted a dislike for the used rest pose between gestures (see Figure 7.4 (step 2)), seeing it as relatively stiff and unnatural. We amend this pose for the next part of this chapter, changing it to the hands hanging by the sides with extended arms (see Figure **??** (2)).

In this first part of the chapter, we used the gesture stroke locations determined by automatic motion segmentation for synthesizing gesture sequences. While is not

practical for a real application, it allowed us to perceptually validate our gesture selection method in a controlled manner. To extend our method to an end-to-end speech-driven gesture generation pipeline in the next part, we combine the proposed method with stroke location prediction from speech. We then evaluate the final speech-to-gesture system against current state-of-the-art machine learning models.

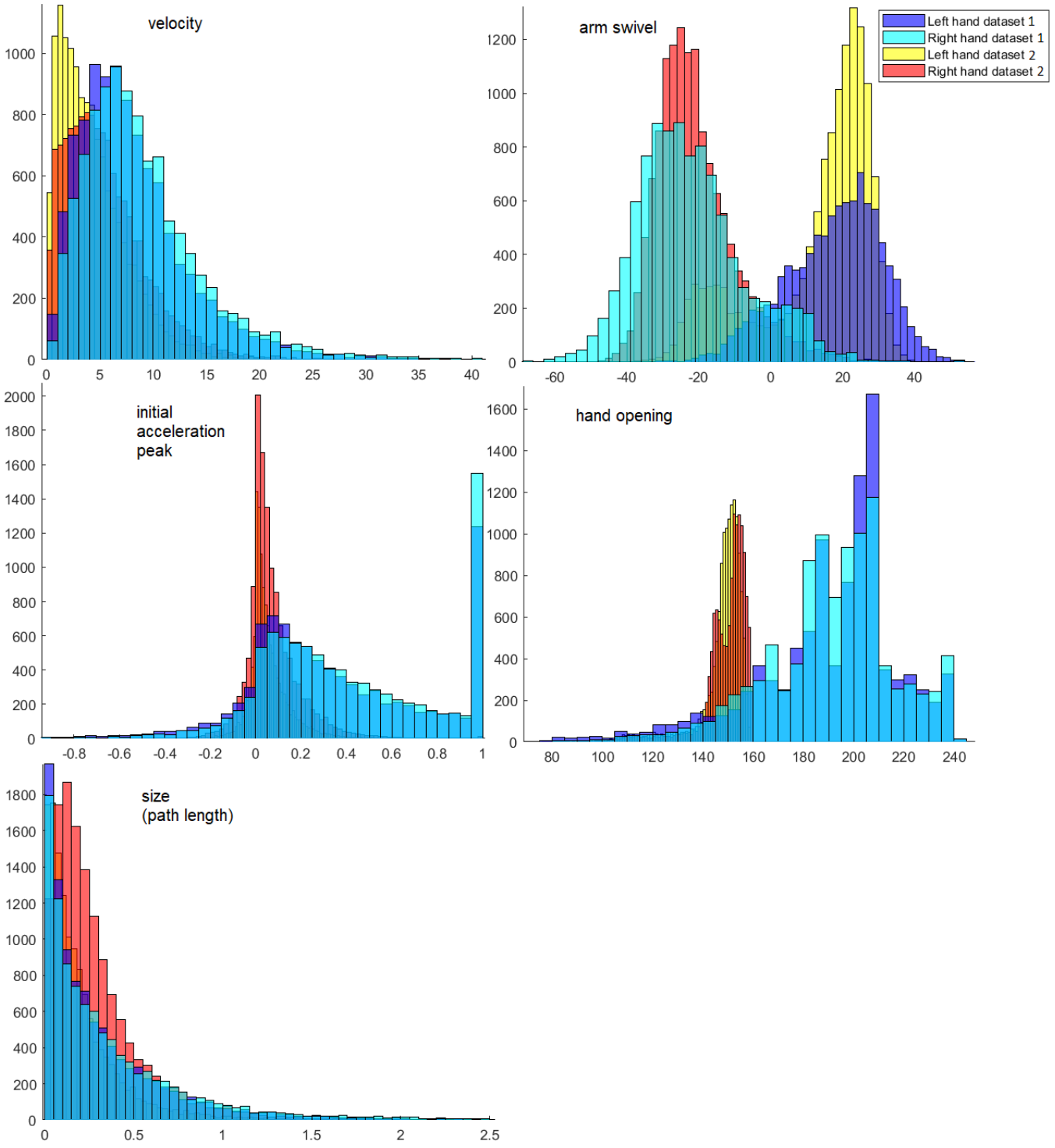


FIGURE 7.7: Distributions of the five gesture parameters in both datasets we used. Note that for arm swivel, increasing the swivel angle (moving the elbow further out) means higher positive values for the left arm, but increasingly negative values for the right arm.

7.4 Gesture timing from speech

For a fully speech-based gesture system, we need to determine when to generate a gesture based only on speech prosody. Through analyzing prosody, we detected points of emphasized speech; we then defined gesture timings to align with speech emphasis. Specifically, we designed a method using the speech pitch the place gestures. We first extracted the pitch tier using Praat. From the pitch tier, we determined the relevant pitch peaks by setting desired prominence and minimum peak distance. These values can be chosen to result in the desired gesture frequency. As we could determine the actual gesture frequency of the speakers within the used datasets through motion analysis, we here set the pitch sensitivity to result in a similar gesture frequency. Both speakers (dataset A and B) present a gesture frequency averaging roughly one gesture every 1.5 seconds. Using an equivalent gesture frequency allows for better comparison to ground truth gesture performance in this study, however, this is a parameter that can be tuned as desired. For example, using a higher gesture frequency elicits higher user ratings of extraversion of a robot [170]. Figure 7.8 given an example of a sequence of detected

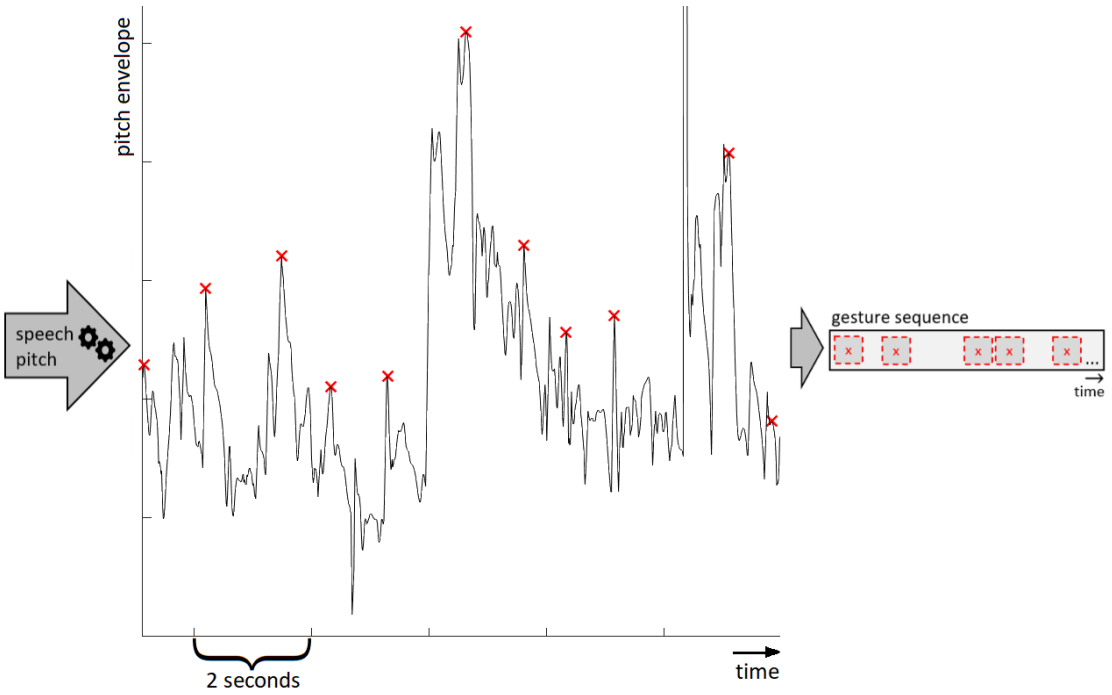


FIGURE 7.8: Detected pitch peaks (marked in red) over a example window of about 11 seconds. Peaks are marked based on their relative prominence under a minimum peak distance constraint. A pitch peak marked the timing of a gesture peak, around which we defined a gesture slot (to be filled later with a gesture from the database).

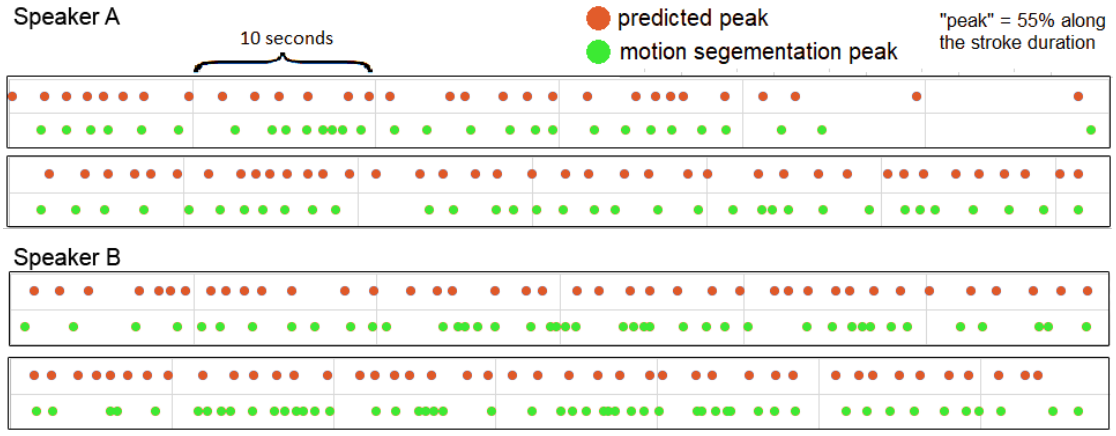


FIGURE 7.9: Comparison of gesture peaks determined by motion segmentation versus by speech pitch analysis. Peak prediction from speech yields plausible rather than exact results. Plotted are two examples of 60 second sequences for each of the two speakers.

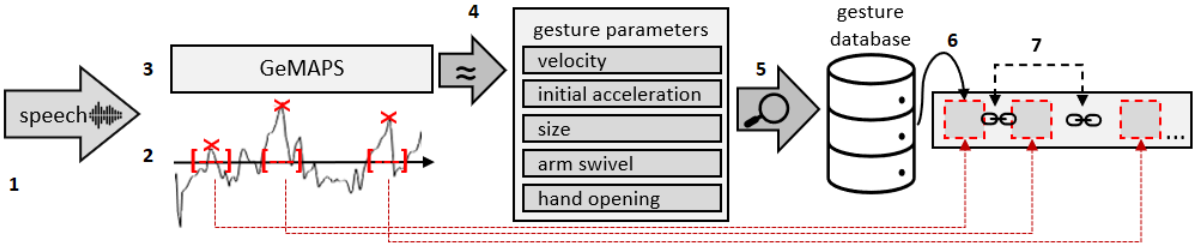


FIGURE 7.10: Overview of our gesture generation system. (1) The system receives as input speech audio. (2) Gesture timings are determined through pitch analysis. (3) GeMAPS prosodic speech features are extracted automatically. (4) From the prosodic features, the desired values for the five gesture parameters are estimated. (5) The database is searched for the gesture with the closest matching parameter values. (6) The best matching gesture is inserted at the desired gesture position. (7) Synthetic preparation and retraction phases are generated to link the gestures in the sequence.

pitch peaks and Figure 7.2 shows the integration of speech-based gesture timing into the system pipeline.

Following the rule that gesture peaks either precede or coincide with the associated speech peak [171], we set gesture timing as follows: Gesture strokes are 55% complete at the determined pitch peaks. A visualization of predicted and true gesture peaks is given in Figure 7.9. We defined the maximum time for a predicted stroke as twice the time to the nearest peak. This time window also defined the speech segment to be used in the following Sec. 7.2 for computing associated gesture parameters and selecting a matching gesture. If the selected gesture was shorter than the time window, we re-aligned the gesture forward to be 55% complete at the pitch peak.

We chose this gesture timing approach for its simplicity and easy reproducibility, however, our gesture generation system allows for combination with any other method

of determining stroke timing.

In Chapter 6, we have seen that gesture size is correlated with duration. When predicting gesture peaks from speech rather than gesture boundaries through motion segmentation, we have less information available to predict gesture size. We therefore amended the parameter weighting from Section 7.2 to reduce the impact of gesture size prediction: $weight_{velocity} = 0.6$, $weight_{acceleration} = 0.8$, $weight_{size} = 1.1$, $weight_{swivel} = 1.3$, $weight_{hand} = 1$.

We evaluated the performance of our speech-to-gesture generation system with two perceptual studies. First, we assessed our method of gesture placement and selection with respect to randomized gestures as well as to the ground truth placement and selection. Next, we compared the performance of our system to state-of-the-art machine learning models.

7.4.1 Perceptual study design

For creating the experiment stimuli, we animated gesture sequences on the GENEA model [132] (see Figure 7.11), using the Unity3D game engine. We chose this model for better comparison to the GENEA results.

For both studies, we distributed an online experiment via the participant-sourcing service Prolific. Participants first read the study instructions and completed a training (detailed in the respective sections below). Following this, each experiment trial consisted of watching a 15 second video clip followed by the question, "How appropriate were the gestures for the speech?", presented with a 7-point Likert scale ranging from "Very bad match" (1) to "Very good match" (7). The phrasing of the rating question was taken from the GENEA gesture generation challenge [132].

Study completion time was approximately 15 minutes. Participants' attention was assessed through content questions: At a random trial number within each quartile of the experiment, a video clip was followed by a multiple choice question about what the speaker said instead of the gesture rating question. Participants achieving less than 50% correct were rejected.

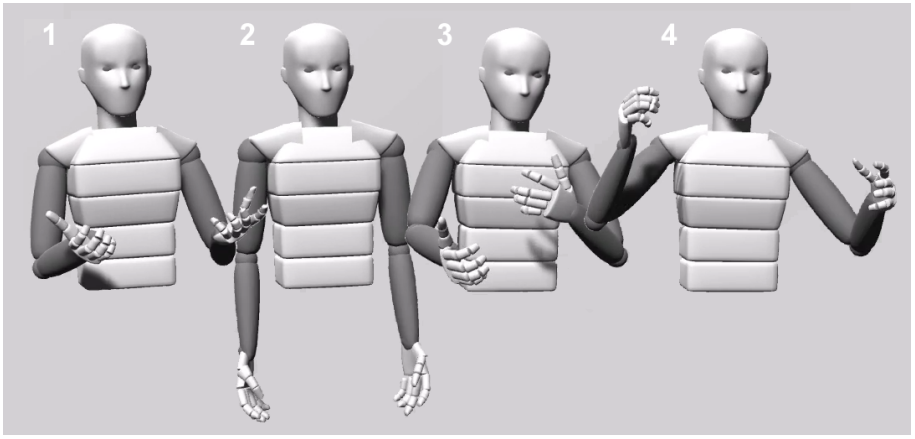


FIGURE 7.11: Example of a generated gesture sequence on the GENEA model.

7.4.2 Experiment II - Baseline

We ran a baseline experiment comparing our method against ground truth (upper bound) and random gesture selection (lower bound).

7.4.2.1 Experiment conditions

Our baseline experiment consisted of three gesture conditions: (1) The ground truth gesture strokes with synthesized preparations, retractions, and transitions, preserving gesture timing (GT-S), (2) random gesture selection with the same overall gesture frequency but not timed to the speech, (3) our method of gesture placement and selection.

In addition to the arm gesture motion, the character displayed some idle torso body swaying and head movements.

7.4.2.2 Perceptual study

Participants first completed a guided training. In the first part of the training, participants were informed that they will watch 2 examples of well-matching speech and gesture, and they were instructed to rate these as such. They were then presented with two ground truth training trials (one for each speaker). Next, in the second part of the training, participants were informed they would now see 2 examples of badly matched speech and gesture, and they were instructed to rate these accordingly. They were then presented with two unmatched (random) training clips (one for each speaker).

Finally, participants were informed about attention checks throughout the experiment and presented with one example.

Participants then completed 30 experiment trials (5 clips for each of the 2 speakers, for the 3 conditions), presented in random order. Each clip contained a different speech segment. For each condition and each participant, the 5 clips for each speaker were selected randomly from a pool of 10-15 clips in order to get a representative sample of generated gesture sequences while minimizing participant fatigue.

A total of 25 participants completed the experiment (12 females, ages - years, $M = 29.9$, $SD = 10.1$), all of whom gave informed consent regarding their participation. Participants represented a wide population sample: 13 different countries were reported as location of residence.

All stimuli can be viewed at
`https://youtube.com/playlist?list=PL04OYHqbFqt0NXC8S0Bkb5Yy1nhKGjCdh`.

7.4.2.3 Results

Statistical analysis of the results of the perceptual experiment was performed by treating the Likert rating scores as ordinal data and fitting a cumulative link model, using clm from the R ordinal package [168].

An ANOVA of the estimated model showed a main effect of condition ($p < .001$), with an effect size measured by Wald Chi Square $\chi^2 = 133.0$.

The ground truth gesture sequence condition was rated significantly higher than both other conditions (both $p < .001$), as expected with a mean rating score of 5.20. Gesture sequences generated with our method were rated significantly higher than random gesture sequences ($p < .01$), with a mean rating score of 3.94 (random: $mean = 3.58$). Results are plotted in Figure 7.12.

7.4.2.4 Discussion

Our method showed better performance than our mismatched *random* condition. This is notable as such a baseline is notoriously hard to beat for automatic gesture generation

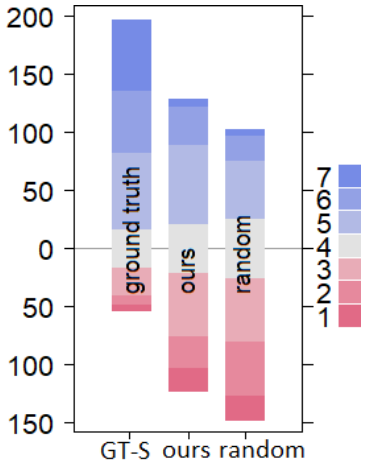


FIGURE 7.12: Stacked bar chart of perceptual ratings for experiment II. Plotted is the frequency of responses for each of the 7 rating scores. (The y-axis is the frequency of responses)

(see the recent state-of-the-art evaluation study GENEA [132], as well as Section 7.4.3 below).

While directly comparing our method to the random condition conflates effects of gesture selection and of timing, this was addressed in the previous perceptual experiment in Section 7.3. There, we compared our method to two baselines, one preserving gesture timing but disregarding expressive parameter match, and one disregarding both timing and match. Our method outperformed both baseline conditions (indeed, preserving gesture timing showed no advantage), suggesting that gesture timing alone was not the reason we here found our method to perform better than the baseline random condition.

7.4.3 Experiment III - Comparative performance evaluation

We compared the performance of our method to current state-of-the-art machine learning models.

7.4.3.1 Experiment conditions

The comparative performance experiment consisted of 9 gesture conditions: (1) The ground truth motion (GT), (2) the ground truth gesture strokes with synthesized preparations, retractions, and transitions (GT-S), (3) mismatched motion, belonging to another speech segment (MM), (4-8) motion generated by the GENEA gesture generation challenge entries (SA-SE) (9) our method.

For conditions GT-S as well as for our method, the character displayed some idle torso body swaying and head movements in addition to the arm gesture motions. All other conditions already contained motion data for these joints.

7.4.3.2 Perceptual study

Participants completed a guided training before starting the experiment. First, participants were informed they would see an example of a very good speech and gesture match and instructed to rate it as such. They were then presented with a ground truth (GT) example trial. Next, instructions informed that they would see an example of badly matching speech and gesture, and they were instructed to rate it as such. An example of mismatched motion (MM) was presented. Participants were then informed that some motions may appear more synthetic while still matching the speech well, followed by an example clip of ground truth gesture strokes with synthetic transitions (GT-S). Next, participants were informed that synthetic motions may show a very bad gesture-speech match, followed by an example clip of mismatched gestures with synthetic transitions.

After the training, participants completed 36 experiment trials (4 clips for each of the 9 conditions), presented in random order. For each condition and each participant, the 4 clips were selected randomly from a large pool of clips, ensuring adequate representation of the variation for each condition while allowing to keep experiment duration short and hence minimizing participant fatigue. Within participant, each clip contained a different speech segment. Because the GENEA challenge included only dataset B, all experiment stimuli were restricted to speech from this dataset.

A total of 30 participants completed the experiment (15 females, ages - years, $M = 26.5$, $SD = 6.9$), all of whom gave informed consent regarding their participation. 10 different countries were reported as location of residence.

All stimuli can be viewed at
`https://youtube.com/playlist?list=PL04OYHqbFqt2ArlWTVaOXh7e9WqWlaJZ7`.

7.4.3.3 Results

An ANOVA of the estimated model showed a main effect of condition ($p < .001$), with an effect size measured by Wald Chi Square $\chi^2 = 234.8$. Both GT and GT-SM were

rated significantly higher than all other conditions (all $p < .001$), interestingly, there was no significant difference between the two. SA was rated significantly lower than all other conditions (all $p < .001$, except $p < .01$ w.r.t. SB). This is comparable to the results reported in Kucherenko et al. [132]. SC was rated significantly higher than SB ($p < .05$), as in Kucherenko et al. [132]. No other differences were significant. Results are visualized in Figure 7.13.

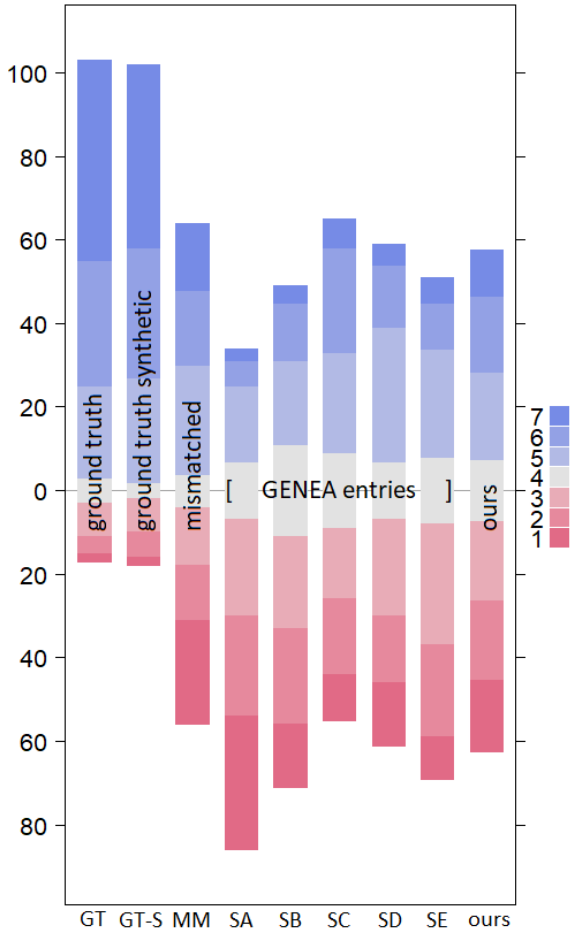


FIGURE 7.13: Stacked bar chart of the frequency of perceptual rating scores for experiment III. Plotted are the 9 conditions in the comparative performance evaluation.

7.4.3.4 Discussion

The results of our third experiment showed that our method produced competitive results to state-of-the-art machine learning approaches with respect to perceived appropriateness. Notably, our method could be distinguished fairly easily from other generated motion through its obvious synthetic transitions between gestures, but was still rated on-par with the more continuous motion of the compared generative approaches.

The speaker in dataset B used in this experiment shows a very animated speaking style, engaging his whole body rather than producing isolated arm gesture motion. Due to the our approach of focusing on generating arm gesture motion with only some auxiliary idle animation for the torso and head, we produced motion closer to the speaking style in dataset 2, in which the speaker remains fairly firmly stanced and producing more isolated arm gesture motion. We therefore think our method produced particularly suitable results for speaker 2. Due to the fact that the model from [132] were not trained on dataset 2, we could not compare stimuli from this data.

Interestingly, the ground truth gestures with synthetic transitions were rated to match the speech equally well as the full motion capture, indicating that matching individual gestures to the speech produces valid results even when this greatly changes the motion style of the speaker.

While we did not compare these conditions directly, The MM condition in this experiment appears to perform better than the random condition in Sec. 7.4.2. Notably, these conditions differed in that MM represented mismatched full motion-capture, whereas the random condition showed synthetic motion blending between gesture strokes. The continuous, fluid motion in MM may have appeared less obviously mismatched to the speech.

7.5 Discussion

We proposed a method for automatic gesture generation from speech audio with realistic gesture form. Previous works on automatic gesture generation from speech often produce motion smoother than natural with poorly defined gesture form, as well as relying on an assumed and implicit speech-gesture relationship. In this work, we generated gestures based on expressive gesture parameters shown to be related to the speech prosody. By selecting gestures from a motion-captured database based on these expressive parameters, we always produced natural and well defined gesture form. Most machine learning approaches generate continuous motion; due to our gesture-by-gesture synthesis approach, our method may be easier to integrate into existing state-based frameworks used by game developers.

7.5.1 Summary

We evaluated our gesture generation system with three perceptual experiments. First, we evaluated our method of gesture selection using expressive parameter matching. For this, we replaced the ground truth gestures either with matched gestures or a random selection. Our results indicated that selecting gestures with our matching method significantly improved perceived speech-gesture match.

In this first experiment, we also investigated the effect of gesture timing and found no significant effect of mismatched gesture timing. This may have been due to the almost continuous speech and frequent gesturing in the used datasets. In cases with longer periods of speech silence, untimed gestures may be perceived worse. This is also supported by the findings of Nirme et al. [96] who report no significant perceptual effect of delaying or advancing gesture timing by 0.5 seconds unless gesture strokes overlapped with speech pauses. Future work could investigate the use of untimed gestures for real-time applications, as no pre-computation is required for gesture selection. This could also be combined with real-time stroke timing prediction methods such as in Levine et al. [111], who set the gestures to begin at syllable peaks. Notably, Fernández-Baena et al. [110] have also reported greater importance of matching gesture quality (measured by stroke intensity) than matching speech timing.

Next, we devised a method for predicting gesture timing from speech and evaluated this in a second perceptual experiment. There, we compared our method to a baseline method selecting random gestures at the same frequency but agnostic to speech emphasis, as well as to the ground truth gestures. Our method of generating gesture sequences for speech proved to outperform unmatched gesture animation.

Finally, in our third perceptual experiment, we compared our method to five current generative machine learning models, as well as to ground truth motion capture, mismatched motion capture, and ground truth gestures with synthetic transitions. The results showed that our method was comparative in performance to the best of the tested generative models, asserting the validity of our proposed approach of gesture generation.

7.5.2 Style control

An advantage of our method of generating and linking individual gestures over continuous generative machine learning models is the possibility of modifying the expression of individual gestures. For example, in a recent work, Sonlu et al. [59] propose a framework for personality expression of virtual agents. The authors modify the Laban Effort and Shape parameters of individual gestures given just their start and end frame, which are known variables within our system. Our method also allows for tuning of gesture frequency, which can be used to modulate perceptions of extraversion [12, 170].

Our method allows for tuning gesture preference by scaling gesture parameters. This could be used for creating gesture behavior specific to a personality, for example, for an extroverted speaker, scaling predicted gesture size up, in order to retain predicted size variation but creating gesture sequences with overall larger gestures. In this study we do not adjust for speaker style or personality other than what may be implicitly expressed through the five gesture parameters. For generating new sequences for one input speaker, we allowed gesture selection from either speaker, resulting in a mix of gestures from both speakers within a sequence. This could potentially create style-mismatches both between gestures, and between gesture and speech. Speaker-specific gesture retrieval is possible with the downside of reducing the amount of available gestures.

7.5.3 Options for system improvement

The motion blending and merging techniques discussed in Section 2.8.2 could be used for more realistic transitions as well as rest pose motion. One alternative technique to combine motion segments into continuous motion are motion graphs, as used in Yang et al. [3] for dyadic conversation gestures. However, building a motion graph for very large datasets can potentially become problematic; Yang et al. [3] note that the motion variety of conversational gesture behavior requires a much larger graph than, for example, was previously used for locomotion. While they build a graph from 30 minutes of motion data, our work uses 20 times that amount. Constructing and searching a motion graph in our case would require questionable computing power. Instead, the appearance of our system's output motion may be improved by creating different rest poses and adding idle motions to the arms and hands between gestures. In our current

motion merging implementation we do not model holds, which are important for gesture co-articulation and emphasis. Moving forward, integrating methods of placing holds is an essential step.

To improve our gesture selection method, the number of gesture parameters used for finding a match could be increased. Any automatically extractable motion measure would be readily integratable in our system. To improve the speech-gesture match, a relationship has to be established between the speech prosody and the new gesture parameter. If the parameter is not computable from speech, it may still be used for biased gesture selection after determining a number of suitable gestures from speech-based parameters, for example in order to achieve a specific gesture style. An avenue for future work is also the classification of gesture types, as in Sadoughi and Busso [172], and association with semantic markers. Using the annotated stroke labels of the datasets used in this work, wrist trajectories could be analyzed to determine simple gesture shapes, such as wiping gestures. Combined with automatic lexical analysis, such as negation tagging, we are interested in exploring the potential for integrating semantically meaningful gesture parameters.

The lack of lexical matching in our current implementation is also one potential reason for disliked gestures within generated sequences. Many of the gestures in the database are iconic gestures: gestures visualizing physical properties and describing the semantic content of the verbalisation. When searching the database for a matching gesture, we only take into account qualitative measures of the gesture (the five gesture parameters), without considering semantic content. Therefore, we sometimes find a gesture match that produces a clear semantic mismatch with the speech. This is different from many machine learning gesture generation models which largely focus on generating beat gestures, gestures without specific meaning but linked to the rhythm and pace of the speech.

7.5.4 Extensibility

Through searching for appropriate gestures by matching motion parameters, we can extend our gesture database with new motion that does not have any audio recordings associated with it, such as the released 20 hours of the *Talking With Hands 16.2M* conversational dataset. This only requires automatic stroke segmentation as utilized

here and presented in Chapter 5, and automatic labelling of the gesture parameters. In this regard, our method differs from Yang et al. [3], who base motion selection on the associated audio segment and therefore require every addition to the database to contain synchronized speech audio annotated with segmentation timings.

Chapter 8

Conclusion

This chapter first provides a summary of this thesis, followed by a discussion of the contributions to current research. We then discuss some limitations of this work.

8.1 Summary

In this thesis, we investigated methods of automatically generating gesture motion solely from speech audio. Through a series of studies, we explored a number of machine learning approaches and their suitability for the speech to gesture training task. We showed that modelling either domain is not sufficient for satisfactory results, rather, the key problem is the mapping *between* the two domains. We have explored the non-deterministic nature of the speech-gesture relationship and developed a gesture generation system based on expressive motion characteristics rather than explicit joint angles or positions as an alternative to current standard approaches in gesture generation research.

First, we created a multimodal dataset of speech and gesture (Chapter 3). We selected two speakers who produce naturally frequent gesture behavior in order to be able to capture significant amounts of suitable data while leaving the actors unaware of the purpose of the recording. We recorded over 10 hours of spontaneous and natural gesture motion, combined from two speakers with differing gesture style.

We analyzed the benefits of prior modelling of the two domains of speech and motion and concluded that this does not make up for the shortcoming of standard

regression training in the problem of modelling the complex relationship between speech and gesture (Chapter 4). Even plausible generated gestures may be penalized heavily when they are numerically far from the ground truth pose sequence, encouraging a convergence to the mean pose, minimizing error across all possibilities, and resulting in lethargic, overly smooth motion around the mean pose.

We addressed this problem in two ways in Chapter 5. First, we tackled the problem of assessing realistic gesture dynamics in order to avoid averaged motion smoothing over accelerations and pauses. Instead, we wanted to model the natural gesture dynamics of periods of emphasis and acceleration, and periods of stillness or pauses: The phases of gesture. Assessing gesture dynamics during model training requires fully automatic gesture phase classification. We therefore hand-annotated large portions of dataset 2 and trained a phase classifier network for the task.

Secondly, we addressed the problems of explicit joint angle error feedback by designing an adversarial training paradigm replacing this error with implicit feedback through a second network deciding how "believable" the gesture output is. Specifically, we define the gesture believability through a series of sub-tasks: Plausible gesture dynamics, as measured through the phase classifier, realistic joint configuration, and smooth and diverse motion. With this, we were the first to use generative adversarial networks for generating 3D gesture motion, and the first to propose a training paradigm with multiple objectives for this task.

We found a clear advantage of using adversarial training over the standard and explicit regression loss, however, we did not find the level of realism satisfactory. Considering the difficulty of modelling gesture motion from speech, we explicitly investigated which gesture characteristics *may* be captured from speech audio alone (Chapter 6). We chose the gesture characteristics to be assessed by drawing from previous research in gesture perception. We determined five motion parameters that have been shown to influence perceptions of gesture in some manner, as well as being automatically extractable from the motion signal, to ensure scalability and re-usability. We trained one model for each gesture parameter and through prediction performance gained a measure of how well the speech signal relates to a given gesture parameter. With a perceptual study, we confirmed that these gesture parameters impact the perceived match of speech and gesture.

Informed by these findings, we proposed a novel gesture generation method based on matching expressive gesture parameters to the speech (Chapter 7). Specifically, our proposed method first estimates matching gesture parameters from speech and then searches our database of gestures for the closest match. We first evaluate the validity of this approach in a perceptual study comparing our method in an ablation manner to two baselines. We showed that selecting gestures based on the estimated parameters significantly improves speech-gesture match. We also showed that correct gesture timing alone is not sufficient for improved appearance. We then extended the proposed approach to a fully speech-based, end-to-end gesture generation pipeline by integrating a method of determining gesture timing from speech. We compared this system against current state-of-the art machine learning approached to gesture generation and found our method to perform competitively.

8.2 Contribution

In general terms, this work contributes a full pipeline for gesture generation from speech audio in a novel hybrid approach of machine learning and database sampling. This work is distinct from other machine learning approaches in several ways. Firstly, our method guarantees defined gesture form through its direct use of motion-captured gestures. Secondly, through the gesture-by-gesture generation approach, gestures can be manipulated individually rather than overall, allowing integration with works on gesture stylization (such as Sonlu et al. [59]). On a more basic level, our approach also allows for tuning of the gesture rate, for example to modulate perceptions of extraversion of the virtual agent [12, 170]. We also believe that through our state machine-like gesture-by-gesture generation manner, our system may be easier to integrate into existing frameworks used by game developers.

In developing a gesture generation system, we have assessed a number of different machine learning approaches and have provided a discussion on their merits. We tested the use of motion as well as language modelling and found no significant benefit for the speech-to-gesture generation problem. We were the first to investigate the use of generative adversarial training for 3D gesture motion and discussed its benefits over standard regression loss for gesture modelling. We reframed the gesture modelling problem in a

new way by training a number of separate networks, each estimating a different expressive gesture parameter, and assessed how well each gesture parameter can be modelled from speech audio.

Another contribution of this work is our method for segmenting gesture motion into its distinct phases. We released our trained classifiers, allowing other researchers to segment new motion recordings. Segmenting gesture motion allows for a deeper, more detailed analysis: Rather than only computing general motion statistics, this allows individual gestures to be assessed for speed, shape, and more. This is useful both for studying the relationship of speech and gesture, as well as assessing gesture generation output: As we have shown in Chapter 6, speech-gesture match is affected by individual gestures' expressive parameters, such as speed and acceleration, and generated gestures should therefore be assessed in a gesture-specific manner rather than assessing general distributions (as in [119, 120]). Segmenting generated output into individual gestures allows computation of per-gesture performance.

Finally, we contribute the to our knowledge largest dataset of speech with synchronized 3D motion. This has already had a major impact on gesture generation research, having been used in numerous works by other researchers, including a recent gesture generation challenge [132]. This dataset has also found use in text-to-speech research, contributing to more natural synthesized speech audio [173, 174]. Additionally, we provide our segmentation of the motion data into individual gestures, making this a unique database for many areas of gesture research.

8.3 Limitations

This work leaves several avenues for improvement. Our results and models are limited to two speakers due to data availability. It is unclear how well the findings generalize to other speakers with different styles of voice and gesture. Additionally, our speakers are both males; male and female motion are perceptually distinct and gesture performance differs between genders (discussed in Section 2.6.3). In this work, we only model upper body motion to reduce complexity, though stance and stepping motions are important in communication, too. Our final solution merges individual gestures into

sequences, however, the transitional motions as well as the rest positions between gestures could be improved for increased realism. Furthermore, when merging individual gesture motions with the simple interpolations used here, we do not achieve naturally diverse co-articulation motions of gesture. If there is time for it, our system retracts the hands to rest between gestures; however, in some cases, a more natural choice is a post-stroke hold plus preparation to the next gesture, or a preparation to the next gesture followed by a pre-stroke hold. We currently do not model such more choices of gesture transitions, but this is necessary for realistic gesture modelling. We did not address real-time gesture generation due to the difficulty of both honoring the gesture-before-speech rule in natural speech and generating speech-matched gestures. We focus our work on monologue-style speaking in order to be able to hone in on one speaker's speech-gesture relationship, however, for a truly communicative agent, dyadic dynamics such as turn-taking need to be addressed. Finally, with current data availability it is difficult to address modelling of semantically meaningful gestures, but these can add greatly to the perceived realism and appropriateness of the gesture behavior, making it an important issue for future work.

Chapter 9

Future Work

There are a number of open problems in the speech-to-gesture mapping task that we did not address due to constraints of time and resources. In this final chapter, we discuss some aspects to be addressed in future work.

One main avenue for future work is the modelling of semantic meaning. One problem hereby is the relatively limited availability of datasets to train this task. Most current machine learning models for automatic gesture generation rely completely on implicit learning of adequate gesturing from training examples. Machine learning models, however, are highly dependent on training set size, requiring many examples to learn a specific "rule". In order to learn to produce semantically meaningful gestures, the training set should contain many examples of a specific gesture co-occurring with the same utterance. This is unlikely unless the speaker purposefully repeats a speech-gesture combination, or a very large dataset is obtained for training. We hope to contribute to the work on this problem with our gesture segmentation in two ways: Firstly, by segmenting individual gestures, automatic shape analysis may become more achievable. If we can, for example, tag all "wipe" gesture in a dataset, modelling when this gesture occurs becomes an easier task. Secondly, even without further shape analysis, we propose that a speech-window-to-gesture rather than the common continuous-speech-to-continuous-motion approach may be a fruitful avenue in future work as this relies to a lesser degree on the implicit learning.

Another area of future work is capturing speaking style. While we recorded two speakers with distinctly different style, we have only quantitatively compared their respective gesture motion to a very limited degree, and we have not used speaking style explicitly in the gesture generation task. Future work could use the desired speaker style as an additional input to the gesture generation system. An open question is the extent and type of cross-speaker gesture comparability, relevant for generalizability and transferability of gesture generation models.

For future investigation of speaking style differences, it will also be essential to include data from more speakers. With the availability of more such data, many interesting areas of gesture research are open, such as personality-based gesture generation and emotion-based gesture modulation. Both of these aspects have been shown in previous work to affect gesture expression, but truly capturing personality or emotion likely requires modelling the modulating impact on gesture *content* too. This also emphasized the importance of the first point, the modelling of semantic meaning as an integral part of gesture behavior. Ultimately, we need to first detect and categorize semantic gestures before attempting to model their complex relationship with speaker style, personality, emotion, and context.

With regard to context, we have here focused on a single speaker in conversational monologue-style, leaving many speaking contexts unaddressed. Firstly, true conversation involves multiple parties, with complex dynamics of taking turns in speaking and listening, using interruptions, responding, and addressing. Future work may include some or all of these dynamics in the modelling of gesture behavior. Secondly, we purposefully focused on relaxed, spontaneous, conversational speech, however, as virtual agents enter a variety of different domains, their behavior has to be adjusted accordingly. For example, a personal companion agent should behave differently than a health care advisor or a virtual teacher. This difference in (gesture) behavior is important not only for eliciting the desired perception of the agent (friendly, knowledgeable, calm, ...) but also for understanding, as a teacher agent for example should be well able to produce helpful deictic gestures, whereas modelling of a companion agent may better focus on interpersonal gestures to prompt responses or show listening and understanding.

Bibliography

[1] Yuyu Xu, Catherine Pelachaud, and Stacy Marsella. Compound gesture generation: A model based on ideational units. In *International Conference on Intelligent Virtual Agents*, pages 477–491. Springer, 2014.

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

[3] Yanzhe Yang, Jimei Yang, and Jessica Hodgins. Statistics-based motion synthesis for social conversations. In *Computer Graphics Forum*, volume 39, pages 201–212. Wiley Online Library, 2020.

[4] Dai Hasegawa, Naoshi Kaneko, Shinichi Shirakawa, Hiroshi Sakuta, and Kazuhiko Sumi. Evaluation of speech-to-gesture generation using bi-directional lstm network. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents*, pages 79–86, 2018.

[5] William Swartout, David Traum, Ron Artstein, Dan Noren, Paul Debevec, Kerry Bronnenkant, Josh Williams, Anton Leuski, Shrikanth Narayanan, Diane Piepol, et al. Ada and grace: Toward realistic and engaging virtual museum guides. In *International Conference on Intelligent Virtual Agents*, pages 286–300. Springer, 2010.

[6] Robert C Hubal, Paul N Kizakevich, Curry I Guinn, Kevin D Merino, and Suzanne L West. The virtual standardized patient. In *Medicine Meets Virtual Reality*, pages 133–138, 2000.

[7] Sin-Hwa Kang, Andrew W Feng, Mike Seymour, and Ari Shapiro. Smart mobile virtual characters: Video characters vs. animated characters. In *Proceedings of*

the Fourth International Conference on Human Agent Interaction, pages 371–374. ACM, 2016.

[8] Maha Salem, Katharina Rohlfing, Stefan Kopp, and Frank Joublin. A friendly gesture: Investigating the effect of multimodal robot behavior in human-robot interaction. In *2011 Ro-Man*, pages 247–252. IEEE, 2011.

[9] Cathy Ennis, Rachel McDonnell, and Carol O'Sullivan. Seeing is believing: Body motion dominates in multisensory conversations. *ACM Transactions on Graphics (TOG)*, 29(4):1–9, 2010.

[10] Regina Pally. A primary role for nonverbal communication in psychoanalysis. *Psychoanalytic inquiry*, 21(1):71–93, 2001.

[11] Susan Goldin-Meadow. The role of gesture in communication and thinking. *Trends in cognitive sciences*, 3(11):419–429, 1999.

[12] Michael Neff, Yingying Wang, Rob Abbott, and Marilyn Walker. Evaluating the effect of gesture and language on personality perception in conversational agents. In *International Conference on Intelligent Virtual Agents*, pages 222–235. Springer, 2010.

[13] Stéphanie Buisine and Jean-Claude Martin. The effects of speech–gesture cooperation in animated agents' behavior in multimedia presentations. *Interacting with Computers*, 19(4):484–493, 2007.

[14] Kirsten Bergmann, Friederike Eyssel, and Stefan Kopp. A second chance to make a first impression? how appearance and nonverbal behavior affect perceived warmth and competence of virtual agents over time. In *International conference on intelligent virtual agents*, pages 126–138. Springer, 2012.

[15] David McNeill. *Hand and mind: What gestures reveal about thought*. University of Chicago press, 1992.

[16] Fritz Heider and Marianne Simmel. An experimental study of apparent behavior. *The American journal of psychology*, 57(2):243–259, 1944.

[17] M Shell. Carnegie mellon university motion capture database, 2012.

[18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36 (7):1325–1339, 2013.

[19] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

[20] Dominik Borer, Dominic Lutz, Robert W Sumner, and Martin Guay. Animating an autonomous 3d talking avatar. In *14th International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing (MCCSIS 2019)*, page 263, 2019.

[21] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion graphs. In *ACM SIGGRAPH 2002 papers*, pages 1–10. 2002.

[22] Simon Clavet and Simon Büttner. Motion Matching - The Road to Next Gen Animation. In *Proc. of Nucl.ai*, 2015.

[23] Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa. Learned motion matching. *ACM Transactions on Graphics (TOG)*, 39(4):53–1, 2020.

[24] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

[25] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 36(4):42, 2017.

[26] Kyungho Lee, Seyoung Lee, and Jehee Lee. Interactive character animation by learning multi-objective control. *ACM Transactions on Graphics (TOG)*, 37(6): 1–10, 2018.

[27] Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020.

[28] Justine Cassell. A framework for gesture generation and interpretation. *Computer vision in human-machine interaction*, pages 191–215, 1998.

[29] Adam Kendon. Some relationships between body motion and speech. *Studies in dyadic communication*, 7(177):90, 1972.

[30] Shuichi Nobe. Where do most spontaneous representational gestures actually occur with respect to speech. *Language and gesture*, 2:186, 2000.

[31] Rachel I Mayberry and Joselynne Jaques. 10 gesture production during stuttered speech: insights into the nature of gesture—speech integration. *Language and Gesture*, 2:199, 2000.

[32] Justine Cassell, David McNeill, and Karl-Erik McCullough. Speech-gesture mismatches: Evidence for one underlying representation of linguistic and nonlinguistic information. *Pragmatics & cognition*, 7(1):1–34, 1999.

[33] Susan Goldin-Meadow and Susan M Wagner. How our hands help us learn. *Trends in cognitive sciences*, 9(5):234–241, 2005.

[34] Todd J. Pruner, Voicu Popescu, and Susan Wagner Cook. The effect of temporal coordination on learning from speech and gesture. In *7th Conference of the International Society for Gesture Studies: Gesture–Creativity–Multimodality (ISGS)*, 2016.

[35] Hans Rutger Bosker and David Peeters. Beat gestures influence which speech sounds you hear. *bioRxiv*, 2020.

[36] Bernard Rimé, Loris Schiaratura, Michel Hupet, and Anne Ghysselinckx. Effects of relative immobilization on the speaker's nonverbal behavior and on the dialogue imagery level. *Motivation and Emotion*, 8(4):311–325, 1984.

[37] Bernard Rimé and Loris Schiaratura. Gesture and speech. *Fundamentals of nonverbal behavior*, pages 239–281, 1991.

[38] Robert M Krauss and Uri Hadar. The role of speech-related arm/hand gestures in word retrieval. *Gesture, speech, and sign*, 93, 1999.

[39] Pierre Feyereisen, Michèle Van de Wiele, and Fabienne Dubois. The meaning of gestures: What can be understood without speech. *European Bulletin of Cognitive Psychology*, 8(3-25):27, 1988.

[40] Robert M Krauss, Palmer Morrel-Samuels, and Christina Colasante. Do conversational hand gestures communicate? *Journal of personality and social psychology*, 61(5):743, 1991.

[41] Robert M Krauss, Robert A Dushay, Yihsiu Chen, and Frances Rauscher. The communicative value of conversational hand gestures. *Journal of experimental social psychology*, 31:533–552, 1995.

[42] Wim Pouw, Lisette de Jonge-Hoekstra, Steven J Harrison, Alexandra Paxton, and James A Dixon. Gesture–speech physics in fluent speech and rhythmic upper limb movements. *Annals of the New York Academy of Sciences*, 2020.

[43] Jana M Iverson and Susan Goldin-Meadow. What's communication got to do with it? Gesture in children blind from birth. *Developmental psychology*, 33(3): 453, 1997.

[44] Lucia Valbonesi, Rashid Ansari, David McNeill, F Quek, Susan Duncan, Karl E McCullough, and R Bryll. Multimodal signal analysis of prosody and hand motion: Temporal correlation of speech and gestures. In *2002 11th European Signal Processing Conference*, pages 1–4. IEEE, 2002.

[45] Kawai Chui. Topicality and gesture in chinese conversational discourse. *LANGUAGE AND LINGUISTICS-TAIPEI-*, 6(4):635, 2005.

[46] Chung-cheng Chiu and Stacy Marsella. Gesture Generation with Low-Dimensional Embeddings. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 781–788, 2014.

[47] Anthony Pak-Hin Kong, Sam-Po Law, Connie Ching-Yin Kwan, Christy Lai, and Vivian Lam. A coding system with independent annotations of gesture forms and functions during verbal communication: Development of a database of speech and gesture (dosage). *Journal of nonverbal behavior*, 39(1):93–111, 2015.

[48] Sotaro Kita, Ingeborg Van Gijn, and Harry Van Der Hulst. Movement Phases in Signs and Co-speech Gestures and Their Transcription by Human Coders. *Gesture and Sign Language in Human-Computer Interaction : International Gesture Workshop Bielefeld*, pages 23–35, 1997.

[49] Sotaro Kita. The temporal relationship between gesture and speech: A study of japanese-english bilinguals. *MS, Department of Psychology, University of Chicago*, 90:91–94, 1990.

[50] David McNeill. A straight path—to where? reply to butterworth and hadar. *Psychological Review*, 96(1):175–179, 1989.

[51] S. Duncan. *Grammatical Form and 'Thinking-for-Speaking' in Mandarin Chinese and English: An Analysis Based on Speech-accompanying Gesture*. PhD thesis, University of Chicago, Unpublished, 1996. Unpublished.

[52] Adam Kendon. *Gesture: Visible action as utterance*. Cambridge University Press, 2004.

[53] Renata Cristina Barros Madeo, Sarajane Marques Peres, and Clodoaldo Aparecido de Moraes Lima. Gesture phase segmentation using support vector machines. *Expert Systems with Applications*, 56:100–115, 2016.

[54] Craig Martell and Joshua Kroll. Corpus-based gesture analysis: an extension of the form dataset for the automatic detection of phases in a gesture. *International Journal of Semantic Computing*, 1(04):521–536, 2007.

[55] Simon Alexanderson, David House, and Jonas Beskow. Automatic annotation of gestural units in spontaneous face-to-face interaction. In *Proceedings of the Workshop on Multimodal Analyses enabling Artificial Agents in Human-Machine Interaction*, pages 15–19, 2016.

[56] Robert Bryll, Francis Quek, and Anna Esposito. Automatic hand hold detection in natural conversation. In *IEEE Workshop on Cues in Communication*, pages 1–4, 2001.

[57] Binyam Gebrekidan Gebre, Peter Wittenburg, and Przemyslaw Lenkiewicz. Towards automatic gesture stroke detection. In *LREC 2012: 8th International Conference on Language Resources and Evaluation*, pages 231–235. European Language Resources Association, 2012.

[58] Michael Kipp, Michael Neff, Kerstin H Kipp, and Irene Albrecht. Towards natural gesture synthesis: Evaluating gesture units in a data-driven approach to gesture

synthesis. In *International Workshop on Intelligent Virtual Agents*, pages 15–28. Springer, 2007.

[59] Sinan Sonlu, Uğur Güdükbay, and Funda Durupinar. A conversational agent framework with multi-modal personality expression. *ACM Transactions on Graphics (TOG)*, 40(1):1–16, 2021.

[60] Irmgard Bartenieff and Dori Lewis. *Body movement: Coping with the environment.* Psychology Press, 1980.

[61] Marion North. *Personality Assessment Through Movement.* Macdonald and Evans, 1972.

[62] Rudolf Laban and Lisa Ullmann. *The Mastery of Movement.* Macdonald and Evans, 1974.

[63] Charles Darwin. *The expression of the emotions in man and animals.* Murray, London, England, 1872. Reprinted, Chicago: University of Chicago Press, 1965.

[64] Paul Ekman and Wallace V Friesen. Detecting deception from the body or face. *Journal of personality and Social Psychology*, 29(3):288, 1974.

[65] Linda A Camras, Jean Sullivan, and George Michel. Do infants express discrete emotions? adult judgments of facial, vocal, and body actions. *Journal of Nonverbal Behavior*, 17(3):171–186, 1993.

[66] Harald G Wallbott. Bodily expression of emotion. *European journal of social psychology*, 28(6):879–896, 1998.

[67] Jacqyln A Levy and Marshall P Duke. The use of laban movement analysis in the study of personality, emotional state and movement style: An exploratory investigation of the veridicality of" body language". *Individual Differences Research*, 1 (1), 2003.

[68] Junya Morita, Yukari Nagai, and Tomoyuki Moritsu. Relations between body motion and emotion: analysis based on laban movement analysis. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 35, 2013.

[69] Arthur Truong, Hugo Boujut, and Titus Zaharia. Laban descriptors for gesture recognition and emotional analysis. *The visual computer*, 32(1):83–98, 2016.

[70] Toru Nakata, Taketoshi Mori, and Tomomasa Sato. Analysis of impression of robot bodily expression. *J. Robotics Mechatronics*, 14(1):27–36, 2002.

[71] Megumi Masuda, Shohei Kato, and Hidenori Itoh. Emotion detection from body motion of human form robot based on laban movement analysis. In *International Conference on Principles and Practice of Multi-Agent Systems*, pages 322–334. Springer, 2009.

[72] Ekaterina P Volkova, Betty J Mohler, Trevor J Dodds, Joachim Tesch, and Heinrich H Bülthoff. Emotion categorization of body expressions in narrative scenarios. *Frontiers in psychology*, 5:623, 2014.

[73] Michael Kipp and Jean-Claude Martin. Gesture and emotion: Can basic gestural form features discriminate emotions? In *2009 3rd international conference on affective computing and intelligent interaction and workshops*, pages 1–8. IEEE, 2009.

[74] Gabriel Castillo and Michael Neff. What do we express without knowing? emotion in gesture. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 702–710, 2019.

[75] Peggy E Gallaher. Individual differences in nonverbal behavior: Dimensions of style. *Journal of personality and social psychology*, 63(1):133, 1992.

[76] Ronald E Riggio and Howard S Friedman. Impression formation: The role of expressive behavior. *Journal of Personality and Social Psychology*, 50(2):421, 1986.

[77] Richard Lippa. The nonverbal display and judgment of extraversion, masculinity, femininity, and gender diagnosticity: A lens model analysis. *Journal of Research in Personality*, 32(1):80–107, 1998.

[78] Michael Argyle. *Bodily communication*. Routledge, 2013.

[79] Mikael Jensen. Personality traits and nonverbal communication patterns. *Int'l J. Soc. Sci. Stud.*, 4:57, 2016.

[80] Jackson Tolins, Kris Liu, Yingying Wang, Jean E Fox Tree, Marilyn Walker, and Michael Neff. Gestural adaptation in extravert-introvert pairs and implications for ivas. In *Intelligent Virtual Agents*, page 484. Springer, 2013.

[81] Harrison Jesse Smith and Michael Neff. Understanding the impact of animated gesture performance on personality perceptions. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017.

[82] Bart GW Craenen, Amol Deshmukh, Mary Ellen Foster, and Alessandro Vinciarelli. Shaping gestures to shape personality: Big-five traits, godspeed scores and the similarity-attraction effect. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2221–2223, 2018.

[83] Yingying Wang, Jean E Fox Tree, Marilyn Walker, and Michael Neff. Assessing the impact of hand motion on virtual character personality. *ACM Transactions on Applied Perception (TAP)*, 13(2):1–23, 2016.

[84] Joseph Bates et al. The role of emotion in believable agents. *Communications of the ACM*, 37(7):122–125, 1994.

[85] Elisabeth André, Thomas Rist, Susanne Van Mulken, Martin Klesen, and Stefan Baldes. The automated design of believable dialogues for animated presentation teams. *Embodied conversational agents*, pages 220–255, 2000.

[86] Kozaburo Hachimura, Katsumi Takashina, and Mitsu Yoshimura. Analysis and evaluation of dancing movement based on lma. In *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, pages 294–299. IEEE, 2005.

[87] Lorenzo Torresani, Peggy Hackney, and Christoph Bregler. Learning motion style synthesis from perceptual observations. In *Advances in neural information processing systems*, pages 1393–1400, 2007.

[88] Andreas Aristidou, Efstathios Stavrakis, Panayiotis Charalambous, Yiorgos Chrysanthou, and Stephania Loizidou Himona. Folk dance evaluation using laban movement analysis. *Journal on Computing and Cultural Heritage (JOCCH)*, 8(4):1–19, 2015.

[89] Durell Bouchard and Norman Badler. Semantic segmentation of motion capture using laban movement analysis. In *International Workshop on Intelligent Virtual Agents*, pages 37–44. Springer, 2007.

[90] Ali-Akbar Samadani, Sarahjane Burton, Rob Gorbet, and Dana Kulic. Laban effort and shape analysis of affective hand and arm movements. In *2013 Humaine Association conference on affective computing and intelligent interaction*, pages 343–348. IEEE, 2013.

[91] Diane Chi, Monica Costa, Liwei Zhao, and Norman Badler. The EMOTE model for effort and shape. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 173–182, 2000.

[92] Liwei Zhao and Norman I Badler. Synthesis and acquisition of laban movement analysis qualitative parameters for communicative gestures. *Technical Reports (CIS)*, page 116, 2001.

[93] Funda Durupinar, Mubbasir Kapadia, Susan Deutsch, Michael Neff, and Norman I Badler. PERFORM: Perceptual Approach for Adding OCEAN Personality to Human Motion Using Laban Movement Analysis. *ACM Transactions on Graphics (TOG)*, 36(1):1–16, 2016.

[94] Björn Hartmann, Maurizio Mancini, and Catherine Pelachaud. Implementing expressive gesture synthesis for embodied conversational agents. In *International Gesture Workshop*, pages 188–199. Springer, 2005.

[95] Yingying Wang and Michael Neff. The influence of prosody on the requirements for gesture-text alignment. In *International Workshop on Intelligent Virtual Agents*, pages 180–188. Springer, 2013.

[96] Jens Nirme, Magnus Haake, Agneta Gulz, and Marianne Gullberg. Motion capture-based animated characters for the study of speech–gesture integration. *Behavior research methods*, pages 1–16, 2019.

[97] Justine Cassell, Hannes Högni Vilhjálmsson, and Timothy Bickmore. BEAT: the Behavior Expression Animation Toolkit. *ACM Transactions on Graphics (TOG)*, pages 477–486, 2001.

[98] Nudge Nudge Wink Wink Cassel. Elements of face-to-face conversation for embodied conversational agents, embodied conversational agents, 2000.

[99] Jina Lee and Stacy Marsella. Nonverbal Behavior Generator for Embodied Conversational Agents. In *International Workshop on Intelligent Virtual Agents*, pages 243–255, 2006.

[100] Marcus Thiebaux, Stacy Marsella, Andrew N Marshall, and Marcelo Kallmann. Smartbody: Behavior realization for embodied conversational agents. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*, pages 151–158. International Foundation for Autonomous Agents and Multiagent Systems, 2008.

[101] Stefan Kopp, Bernhard Jung, Nadine Lessmann, and Ipke Wachsmuth. Max-a multimodal assistant in virtual reality construction. *KI*, 17(4):11, 2003.

[102] Sri Rama Kartheek Kappagantula, Nicoletta Adamo-Villani, Meng-Lin Wu, and Voicu Popescu. Automatic deictic gestures for animated pedagogical agents. *IEEE Transactions on Learning Technologies*, 13(1):1–13, 2019.

[103] Stacy Marsella, Yuyu Xu, Margaux Lhommet, Andrew Feng, Stefan Scherer, and Ari Shapiro. Virtual character performance from speech. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 25–35, 2013.

[104] Geneviève Calbris. *Elements of meaning in gesture*, volume 5. John Benjamins Publishing, 2011.

[105] Margot Lhommet and Stacy Marsella. Metaphoric gestures: Towards grounded mental spaces. In *International Conference on Intelligent Virtual Agents*, pages 264–274. Springer, 2014.

[106] Brian Ravenet, Chloé Clavel, and Catherine Pelachaud. Automatic Nonverbal Behavior Generation from Image Schemas. In *Proceedings of the 17th international conference on autonomous agents and multiagent systems*, pages 1667–1674, 2018.

[107] Kirsten Bergmann, Stefan Kopp, and Friederike Eyssel. Individualized gesturing outperforms average gesturing–Evaluating gesture production in virtual humans. In *International Conference on Intelligent Virtual Agents*, pages 104–117. Springer, 2010.

[108] Kirsten Bergmann and Stefan Kopp. GNetIc - Using bayesian decision networks for iconic gesture generation. In *International Workshop on Intelligent Virtual Agents*, pages 76–89. Springer, 2009.

[109] Kirsten Bergmann, Sebastian Kahl, and Stefan Kopp. Modeling the semantic coordination of speech and gesture under cognitive and linguistic constraints. In *International Workshop on Intelligent Virtual Agents*, pages 203–216. Springer, 2013.

[110] Adso Fernández-Baena, Raúl Montaño, Marc Antonijoan, Arturo Roversi, David Miralles, and Francesc Alías. Gesture synthesis adapted to speech emphasis. *Speech communication*, 57:331–350, 2014.

[111] Sergey Levine, Christian Theobalt, and Vladlen Koltun. Real-time prosody-driven synthesis of body language. In *ACM SIGGRAPH Asia 2009 papers*, pages 1–10. 2009.

[112] Elif Bozkurt, Yücel Yemez, and Engin Erzin. Multimodal analysis of speech and arm motion for prosody-driven synthesis of beat gestures. *Speech Communication*, 85:29–42, 2016.

[113] Sergey Levine, Philipp Krähenbühl, Sebastian Thrun, and Vladlen Koltun. Gesture controllers. In *ACM SIGGRAPH 2010 papers*, pages 1–11. 2010.

[114] Chung-Cheng Chiu and Stacy Marsella. How to train your avatar: A data driven approach to gesture generation. In *International Workshop on Intelligent Virtual Agents*, pages 127–140. Springer, 2011.

[115] Chung-Cheng Chiu, Louis-Philippe Morency, and Stacy Marsella. Predicting Co-verbal Gestures: A Deep and Temporal Modeling Approach. In *International Conference on Intelligent Virtual Agents*, pages 152–166. Springer, 2015.

[116] Dario Pavllo, David Grangier, and Michael Auli. Quaternet: A quaternion-based recurrent model for human motion. *arXiv preprint arXiv:1805.06485*, 2018.

[117] Zimo Li, Yi Zhou, Shuangjiu Xiao, Chong He, and Hao Li. Auto-conditioned lstm network for extended complex human motion synthesis. *arXiv preprint arXiv:1707.05363*, 3, 2017.

[118] Kenta Takeuchi, Dai Hasegawa, Shinichi Shirakawa, Naoshi Kaneko, Hiroshi Sakuta, and Kazuhiko Sumi. Speech-to-Gesture Generation: A Challenge in Deep Learning Approach with Bi-Directional LSTM. In *Proceedings of the 5th International Conference on Human Agent Interaction*, pages 365–369, 2017.

[119] Taras Kucherenko, Dai Hasegawa, Gustav Eje Henter, Naoshi Kaneko, and Hedvig Kjellström. Analyzing input and output representations for speech-driven gesture generation. In *IVA '19: International Conference on Intelligent Virtual Agents (IVA '19)*, 2019.

[120] Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexandersson, Iolanda Leite, and Hedvig Kjellström. Gesticulator: A framework for semantically-aware speech-driven gesture generation. In *Proceedings of the 2020 International Conference on Multimodal Interaction*, pages 242–250, 2020.

[121] Emad Barsoum, John Kender, and Zicheng Liu. Hp-gan: Probabilistic 3d human motion prediction via gan. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1418–1427, 2018.

[122] Jogendra Nath Kundu, Maharshi Gor, and R Venkatesh Babu. BiHMP-GAN: Bidirectional 3D Human Motion Prediction GAN. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8553–8560, 2019.

[123] Najmeh Sadoughi and Carlos Busso. Novel realizations of speech-driven head movements with generative adversarial networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6169–6173. IEEE, 2018.

[124] Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. Learning individual styles of conversational gesture. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3497–3506, 2019.

[125] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[126] Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020.

[127] Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. Style-controllable speech-driven gesture synthesisusing normalising flows. In *EUROGRAPHICS 2020*, 2020.

[128] Uttaran Bhattacharya, Nicholas Rewkowski, Abhishek Banerjee, Pooja Guhan, Aniket Bera, and Dinesh Manocha. Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. *arXiv preprint arXiv:2101.11101*, 2021.

[129] Jan Ondras, Oya Celiktutan, Paul Bremner, and Hatice Gunes. Audio-driven robot upper-body motion synthesis. *IEEE Transactions on Cybernetics*, 2020.

[130] Taras Kucherenko, Dai Hasegawa, Naoshi Kaneko, Gustav Eje Henter, and Hedvig Kjellström. Moving fast and slow: Analysis of representations and post-processing in speech-driven automatic gesture generation. *arXiv preprint arXiv:2007.09170*, 2020.

[131] Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Speech gesture generation from the trimodal context of text, audio, and speaker identity. *ACM Transactions on Graphics (TOG)*, 39(6): 1–16, 2020.

[132] Taras Kucherenko, Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, and Gustav Eje Henter. The genea challenge 2020: Benchmarking gesture-generation systems on common data. 2020.

[133] Pieter Wolfert, Nicole Robinson, and Tony Belpaeme. A Review of Evaluation Practices of Gesture Generation in Embodied Conversational Agents. *arXiv preprint arXiv:2101.03769*, 2021.

[134] Najmeh Sadoughi and Carlos Busso. Speech-driven animation with meaningful behaviors. *Speech Communication*, 110:90–100, 2019.

[135] Unai Zabala, Igor Rodriguez, José María Martínez-Otzeta, Itziar Irigoien, and Elena Lazkano. Quantitative analysis of robot gesticulation behavior. *Autonomous Robots*, pages 1–15, 2021.

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

[137] Chien-Ming Huang and Bilge Mutlu. Modeling and evaluating narrative gestures for humanlike robots. In *Robotics: Science and Systems*, pages 57–64, 2013.

[138] Chien-Ming Huang and Bilge Mutlu. Learning-based modeling of multimodal behaviors for humanlike robots. In *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 57–64. IEEE, 2014.

[139] Bowen Wu, Chaoran Liu, Carlos Toshinori Ishi, and Hiroshi Ishiguro. Modeling the conditional distribution of co-speech upper body gesture jointly using conditional-gan and unrolled-gan. *Electronics*, 10(3):228, 2021.

[140] Izidor Mlakar, Zdravko Kačič, and Matej Rojc. TTS-driven synthetic behaviour-generation model for artificial bodies. *International Journal of Advanced Robotic Systems*, 10(10):344, 2013.

[141] Eli Shlizerman, Lucio Dery, Hayden Schoen, and Ira Kemelmacher-Shlizerman. Audio to Body Dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7574–7583, 2018.

[142] Justine Cassell, Timothy Bickmore, Mark Billinghurst, Lee Campbell, Kenny Chang, Hannes Vilhjálmsson, and Hao Yan. Embodiment in conversational interfaces: Rea. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 520–527, 1999.

[143] David DeVault, Ron Artstein, Grace Benn, Teresa Dey, Ed Fast, Alesia Gainer, Kallirroi Georgila, Jon Gratch, Arno Hartholt, Margaux Lhommet, et al. SimSensei Kiosk: A virtual human interviewer for healthcare decision support. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, pages 1061–1068, 2014.

[144] Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. Transfer learning for low-resource neural machine translation. *arXiv preprint arXiv:1604.02201*, 2016.

[145] Long D Nguyen, Dongyun Lin, Zhiping Lin, and Jiuwen Cao. Deep cnns for microscopic image classification by exploiting transfer learning and feature concatenation. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE, 2018.

[146] Hong-Wei Ng, Viet Dung Nguyen, Vassilios Vonikakis, and Stefan Winkler. Deep learning for emotion recognition on small datasets using transfer learning. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*, pages 443–449, 2015.

[147] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent Network Models for Human Dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4346–4354, 2015.

[148] Julieta Martinez, Michael J Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2891–2900, 2017.

[149] Li Deng, Jinyu Li, Jui-Ting Huang, Kaisheng Yao, Dong Yu, Frank Seide, Michael Seltzer, Geoff Zweig, Xiaodong He, Jason Williams, et al. Recent advances in deep learning for speech research at microsoft. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 8604–8608. IEEE, 2013.

[150] Chuang Ding, Lei Xie, and Pengcheng Zhu. Head motion synthesis from speech using deep neural networks. *Multimedia Tools and Applications*, 74(22):9871–9888, 2015.

[151] Florian Eyben, Felix Weninger, Florian Gross, and Björn Schuller. Recent developments in opensmile, the munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 835–838. ACM, 2013.

[152] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-RNN: Deep Learning on Spatio-Temporal Graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 5308–5317, 2016.

[153] Yi Zhou, Zimo Li, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. 2018.

[154] Michael Kipp. Anvil-a generic annotation tool for multimodal dialogue. In *Seventh European Conference on Speech Communication and Technology*, 2001.

[155] Ron Artstein and Massimo Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

[156] Fajrian Yunus, Chloé Clavel, and Catherine Pelachaud. Gesture class prediction by recurrent neural network and attention mechanism. In *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*, pages 233–235. ACM, 2019.

[157] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[158] Ishan Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative multi-adversarial networks. *arXiv preprint arXiv:1611.01673*, 2016.

[159] Isabela Albuquerque, João Monteiro, Thang Doan, Breandan Considine, Tiago Falk, and Ioannis Mitliagkas. Multi-objective training of generative adversarial networks with multiple discriminators. *arXiv preprint arXiv:1901.08680*, 2019.

[160] Ylva Ferstl, Michael Neff, and Rachel McDonnell. Adversarial gesture generation with realistic gesture phasing. *Computers & Graphics*, 2020.

[161] Florian Eyben, Klaus R Scherer, Björn W Schuller, Johan Sundberg, Elisabeth André, Carlos Busso, Laurence Y Devillers, Julien Epps, Petri Laukka, Shrikanth S Narayanan, et al. The Geneva Minimalistic Acoustic Parameter Set (GeMAPS) for Voice Research and Affective Computing. *IEEE transactions on affective computing*, 7(2):190–202, 2015.

[162] Michael Neff and Yejin Kim. Interactive editing of motion style using drives and correlations. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 103–112, 2009.

[163] Ari Shapiro, Petros Faloutsos, and Victor Ng-Thow-Hing. Dynamic animation and control environment. In *Proceedings of graphics interface 2005*, pages 61–70. Canadian Human-Computer Communications Society, 2005.

[164] Ted Shawn. *Every little movement: A book about François Delsarte, the man and his philosophy, his science and applied aesthetics, the application of this science to the art of the dance, the influence of Delsarte on American dance.* Printed by the Eagle Print. and Binding Co., 1963.

[165] Wim Pouw, Steven J Harrison, and James A Dixon. Gesture–speech physics: The biomechanical basis for the emergence of gesture–speech synchrony. *Journal of Experimental Psychology: General*, 2019.

[166] Daniel P Loehr. Temporal, structural, and pragmatic synchrony between intonation and gesture. *Laboratory phonology*, 3(1):71–89, 2012.

[167] Arno Hartholt, David Traum, Stacy C Marsella, Ari Shapiro, Giota Stratou, Anton Leuski, Louis-Philippe Morency, and Jonathan Gratch. All together now. In *International Workshop on Intelligent Virtual Agents*, pages 368–381. Springer, 2013.

[168] Rune Haubo Bojesen Christensen. ordinal—regression models for ordinal data. *R package version*, 28:2015, 2015.

[169] Ylva Ferstl and Rachel McDonnell. Investigating the use of recurrent motion modelling for speech gesture generation. In *IVA '18: International Conference on Intelligent Virtual Agents (IVA '18)*, pages 93–98, 2018.

[170] Heeyoung Kim, Sonya S Kwak, and Myungsuk Kim. Personality design of sociable robots by control of gesture design factors. In *RO-MAN 2008-The 17th IEEE International Symposium on Robot and Human Interactive Communication*, pages 494–499. IEEE, 2008.

[171] Adam Kendon. Gesticulation and speech: Two aspects of the process of utterance. *The relationship of verbal and nonverbal communication*, 25(1980):207–227, 1980.

[172] Najmeh Sadoughi and Carlos Busso. Retrieving target gestures toward speech driven animation with meaningful behaviors. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 115–122, 2015.

[173] Éva Székely, Gustav Eje Henter, Jonas Beskow, and Joakim Gustafson. Breathing and speech planning in spontaneous speech synthesis. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7649–7653. IEEE, 2020.

[174] Simon Alexanderson, Éva Székely, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. Generating coherent spontaneous speech and gesture from text. In *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents*, pages 1–3, 2020.