THE UNIVERSITY *of* EDINBURGH

Speech wave-form Driven Motion Synthesis For Embodied Agents

JinHong Lu

Supervisors: Dr. H. Shimodaira

Doctor of Philosophy

School of Informatics

University of Edinburgh

2023

Acronyms

- AE Auto-encoder
- AMMSE Affine minimum mean square error estimator
- AMPD Automatic multiscale-based peak detection
- ASR Automatic speech recognition
- BA baseline model trained with audio
- Bi-LSTM/BLSTM Bi-direction long short term memory
- BT baseline model trained with text
- CCA Canonical correlation analysis
- CLDNN Convolutional long short-term memory deep neural network
- CorrNet Correlated neural network
- DCCAE Deep canonical correlated auto-encoder
- DNN Deep neural network
- EEG Electroencephalogram
- EMA Electro-magnetic articulograph
- FNN Feedforward neural network
- GAN Generative adversarial network
- GRU Gated recurrent unit
- HMM Hidden Markov model
- KL Kullback–Leibler
- LPC Linear predictive Coefficients

- LSP Linear spectral pairs
- LSTM Long short term memory
- MAE multimodel auto-encoder
- MCEP Mel-cepstral coefficients
- MGCEP Mel-generalized cepstral coefficients
- MFCC Mel-frequency cepstrum
- MLPG Maximum likelihood parameter generation
- MOS Mean opinion score
- MSE Mean square error
- MUSHRA Multiple stimuli with hidden reference and anchor
- NMSE Nomarlised mean square error
- RNN Recurrent neural network
- SOTA State of the art
- TTS Text to speech
- TWV Term weighted value

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(JinHong Lu)

Abstract

The main objective of this thesis is to synthesise motion from speech, especially in conversation. Based on previous research into different acoustic features or the combination of them were investigated, no one has investigated in estimating head motion from waveform directly, which is the stem of the speech. Thus, we study the direct use of speech waveform to generate head motion. We claim that creating a task-specific feature from waveform to generate head motion leads to better performance than using standard acoustic features to generate head motion overall. At the same time, we completely abandon the handcrafted feature extraction process, leading to more effectiveness. However, there are a few problems if we would like to apply speech waveform, 1) high dimensional, where the dimension of the waveform data is much higher than those common acoustic features and thus making the training of the model more difficult, and 2) irrelevant information, which refers to the full information in the original waveform implicating potential cumbrance for neural network training. To resolve these problems, we applied a deep canonical correlated constrainted auto-encoder (DCCCAE) to compress the waveform into low dimensional and highly correlated embedded features with head motion. The estimated head motion was evaluated both objectively and subjectively. In objective evaluation, the result confirmed that DCCCAE enables the creation of a more correlated feature with the head motion than standard AE and other popular spectral features such as MFCC and FBank, and is capable of being used in achieving state-of-the-art results for predicting natural head motion with the advantage of the DCCCAE. Besides investigating the representation learning of the feature, we also explored the LSTM-based regression model for the proposed feature. The LSTM-based models were able to boost the overall performance in the objective evaluation and adapt better to the proposed feature than MFCC. MUSHRA-liked subjective evaluation results suggest that the animations generated by models with the proposed feature were chosen to be better than the other models by the participants of MUSHRA-liked

test. A/B test further that the LSTM-based regression model adapts better to the proposed feature. Furthermore, we extended the architecture to estimate the upper body motion as well. We submitted our result to GENEA2020 and our model achieved a higher score than BA in both aspects (human-likeness and appropriateness) according to the participant's preference, suggesting that the highly correlated feature pair and the sequential estimation helped in improving the model generalisation.

Acknowledgements

First, I would like to give great thanks to my supervisor, Dr Hiroshi Shimodaira. Hiroshi is a kind and patient elder, and he has not only encouraged me throughout my PhD time but also coached me since my undergraduate degree. I still remember during my undergraduate thesis, I was a new kid on the topic of speech motion generation when I met him in his office for the first meeting. Hiroshi explained clearly in a few sentences and suggested relevant papers for me to understand and address the topic in a short time, which enabled me to be more sophisticated in doing research and have good outcomes in my following study time. I admire that Hiroshi always provides valuable discussions during our weekly meetings, where I learned a lot in both the aspect of practical work and the aspect of novel ideas. This thesis would not have been possible without Hiroshi's guidance and support.

I am grateful to the people in the Centre for Speech Technology Research (CSTR). CSTR hosts research sharing and discussions weekly, and their unique insights into the research inspired me to think about my research in a broader context.

I would like to thank my friends in the UK for their support. I especially would like to thank MengTing Chai, HanXi Qiao, PinZhen Chen, HongJian Li, LiXian Li, ZhiHua Li, XiaoDong Xi, HongQiu Wang, DeJian Zhang and Zheng Zhao. I would cherish the time we had spent together.

Finally, I am grateful for the support of my family throughout the journey. They had been giving all they could provide without words when I decided to go for a PhD. Without them, I could not be what I am now.

List of Figures

List of Tables

Table of Contents

Chapter 1

Introduction

1.1 Motivation

As the world has been rapidly changing, human-to-human communication is no longer the only communication pair. Nowadays, humans have also been communicating and interacting with computers, robots and smart phones. In some ways, these emerging communication pairs are even more prevalent than human-to-human communication. In this regard, the most recent example is the artificially intelligent robot 'Sophia' created and programmed by Hanson Robotics (GRESHKO [2018]). She is the first artificial intelligence (AI) robot citizen to be recognised with a citizenship of Saudi Arabia. Sophia is capable of simulating human-like facial features and expressions and processing massive social data gathered from interactions; moreover, she can hold eye contact, recognise faces and understand human speech. Sophia's ability to listen, speak and move possibly demonstrates the minimum level of human-like interaction. We believe that these three factors are the keys of human-likeness for robots, and they can not become better without any of those factors.

Nowadays, regardless of the communication object, speech is the first medium that comes to mind; however, speech is only one of the most commonly used communica-

tion modes. In prehistoric times, human beings used gestures to communicate, express intentions and exchange ideas (Pollick and de Waal [2007]). Interestingly, we still use this mode of communication nowadays as well; for example, we nod our heads to express agreement with people's opinions during conversations, guiding the direction with gestures and so on. Overall, the communication medium can be categorised as verbal (speech) and nonverbal (gesture). Both communication mediums differ in the following aspects (Riggio and Riggio [2012]):

- Single channel (spoken word, written word) VS multiple channel (motion, tone of voice)
- Shared code (language) VS impressionistic (unique interpretation)
- Precise VS spontaneous

A fascinating aspect of nonverbal communication is its historical track, which can be traced back to human ancestors, demonstrating the important role of the nonverbal communication channel in the history of human development. Thus, we believe that nonverbal communication should not be neglected in research on human-computer interaction towards the goal of human-likeness.

The definition of nonverbal communication can be noted in both face-to-face and mediated communication (Patterson [2017]). Thus, nonverbal communication should not be restricted to gestures and head nodding, which we have provided as examples above, but should also include facial expressions, paralinguistics such as loudness or tone of voice, body language, proxemics or personal space, eye gaze, haptics (touch), apperance, and artifacts (Anderson [2006]). Most of the nonverbal signs happen automatically and often outside of awareness, connecting people within the conversation by providing information, regulating interaction, expressing intimacy, exercising influence and managing impressions (Patterson [2017]). Nevertheless, this channel of communication can further be defined as expressing emotions, conveying interpersonal

attitudes such as friendliness, anomosity, or dominance, regulating affect, regulating turn taking between people in conversation and facilitating one's own speech production (Hall [2001]). However, in the scope of this thesis, we limit the investigation to motion happening in conversation only. That is because this scenario involves the most interaction between speech and motion.

Different types of motion act as nonverbal signals within the conversation, including the following: posture, hand gestures, lip movement, facial expressions (typically involving expressions of emotions) and head motions. We distinguish the following types of motions in our own definition:

- Gesticulation: movements that involve hands and arms only
- Lip: mouth movements only
- Head: movements of the head
- Body: upper body movements, but exclude hand, arm and head

Each type of motion plays a different role within the conversation. Gesticulation can be used to recall words and shape our thoughts (Goldin-Meadow and Alibali [2013]). Through gesticulation, we can express our thoughts, emotions and intentions in conversations (McNeill [1992]). Gesticulation can also involve relevant aspects of our immediate environment when we manually point at things to guide the attention of our addressee (Peeters et al. [2015]). In particular, lip expressions are great exhibitors of the seven universal micro expressions and can help map micro-expressions of emotion (Matsumoto and Hwang [2011]). Moreover, lip movement can be beneficial for speech comprehension when the acoustic signal is degraded (Peelle and Sommers [2015]; Sumby and Pollack [1954]; van Wassenhove et al. [2005]; Zion-Golumbic and Schroeder [2012]). Similarly, head motion also plays a communicative role within a conversational as it can control the talking turn sequence or convey specific meaning (Heylen [2005]). Furthermore, head motion is also important to show and establish

a connection by providing supportive backchannel cues while listening to the speaker (Gratch et al. [2006]; Huang et al. [2011]). Additionally, head motion can represent the mood of the speaker (Busso et al. [2007a,b]) and express uncertainty (Marsi and van Rooden [2007]) because it carries semantic meaning, important conversational clues, and expression (McClave [2000]). With so many functionalities stated, this thesis cannot go cover the vast variety of motion. Lip motion is more correlated to facial expression, especially to emotions, which is another research domain. Moreover, head/gesture/body motions require a planer to synchronise with speech (Fig 1.1(c)) in production, whereas lip motions are produced in between action and message generators. The differences in the level of production between lip and head/gesture/body motions indicate that the higher the level of the production is, the more difficult the generation task is. As explore a higher level of motion production, lip motion is then excluded from the scope of this thesis.

On a further note, the production of motion in conversation should be discussed. Figure 1.1 shows several models (Krauss and Hadar [1999]; de Ruiter [2000]; Kita and Özyürek [2003]), where the dependencies for motion during speech are proposed by some researchers. The top level of production shown in the diagram is written as 'working memory', which is used to allocate the speech and motion works to the next level. The main difference between the three models in the diagram is that Figures 1.1(b) and (c) designate a planner for speech and motion, but that is not the case for Figure 1.1(a). Figure 1.1(a) indicates that the 'gesture lead' phenomenon arises since gesture, unlike speech, does not require linguistic processing (Mcneill [1987]). Whereas Figure 1.1(b) and (c) suggest that the common origin of gesture and speech is located on the pre-semantic level of communicative intention, which activates both abstract propositional representations and motoric representations (Morrel-Samuels and Krauss [1992]). In the original paper, Krauss and Hadar [1999] argued that the same information from conceptualiser as the input to the formular, it would be difficult to see

how gestural information could facilitate lexical retrieval. However, de Ruiter [2000] stated that the output of the Conceptualiser would be a representation called the preverbal message, which contains a propositional representation of the content of the speech. The reason for the concept difference between the two researchers is that Krauss and Hadar [1999] assumed 'gesture lead' in the speech-motion production and the output of the Conceptualiser would be gesture information, but de Ruiter [2000] expected the gestures would be initialed by a process that was in some way linked to the speaking process. Another difference for the Conceptualizer in Figure 1.1(b) displays feedback communication links; however, Figure 1.1(c) shows that the interval duration by which gesture precedes speech, as well as the duration of gesture, appear to be a function of how familiar the lexical affiliate is to the speaker (Morrel-Samuels and Krauss [1992]). This interaction difference is caused by the splitting of the Conceptualizer in Kita and Özyürek [2003]. Kita and Özyürek [2003] suggested to split the Conceptualizer into two halves, called communication planner and message generator. The communication planner was expected to generate "communicative intention" and fulfil equivalent functions to Levelt [1993]'s "macro-planning" (i.e., rough decision on information to be expressed, rough ordering of parts of the information for expression, and selection of appropriate speech acts). In addition, it determined which modalities of expression should be involved. The second half is the Message Generator, which fulfilled functions equivalent to Levelt [1993]'s "micro-planning" (i.e., formulating a proposition to be verbally formulated while taking into account both the communicative goal of an utterance and the discourse context). The interaction between the gesture planner and the Conceptualizer in de Ruiter [2000] is actually moved to lower level in Kita and Özyürek [2003] as the speech information was split in message generator.

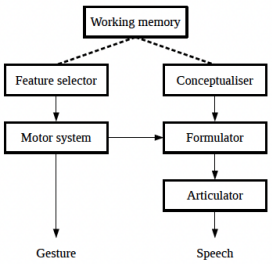
In our thesis, our scope is to investigate the motion that occurs during the conversation. Thus, Krauss and Hadar [1999]'s model is not the choice for us as the assumption of the 'gesture lead' motion production differs from our scope, which is speech-driven

motion synthesis. Moreover, Kita and Özyürek [2003]'s model is a better choice than de Ruiter [2000]'s because with the advantages of splitting the Conceptualizer, we now could only focus the interaction of the action generator as we would not explore any speech-related tasks in the formulator. However, it still needs to be adapted to include some additional dependencies based on the personality of the speaker and the properties of the motion that do not differ between speakers (shown in Figure 1.1(d)). Adding these two boxes makes the model now to be more speaker-dependent.

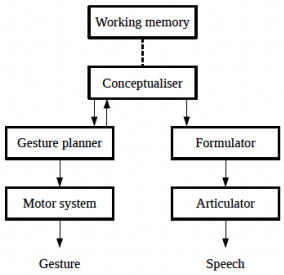
1.2 Objective

Within the thesis, we firstly aim to find a method to generate head motion from speech as the head is directly affected by the articulator while speaking. Second, we try to generate gesticulation and body motion for deeper level of interaction during the conversation. Furthermore, we also demonstrate the mapping's validity and necessity, with the caveat that this does not include semantic motion.

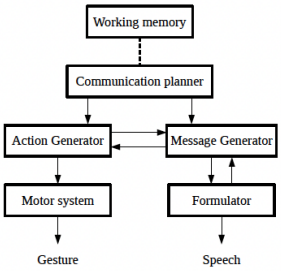
We particularly focus on finding a suitable neural network to map speech waveform to natural head motion, an end-to-end system. In research on head motion, no study has directly mapped waveform to head motion before. This is mainly because of (1) the high dimensionality of raw waveform signals, which slows down the training of neural networks and requires high capacity in the hardware support, (2) a large amount of irrelevant information to predict head motion, which hinders the training of neural networks. Another reason to argue the advantage of using waveform over those hand-crafted features such as log Mel filterbank, Mel-frequency cepstral coefficients (MFCC) is to avoid potential information loss which might result in suboptimal performance (Vieting et al. [2021]). This lost information during the extraction process, such as prosodic information, is helpful in estimating head motion Kuratate et al. [1999]. How the prosodic information (e.g. F0) links to head motion will be discussed in



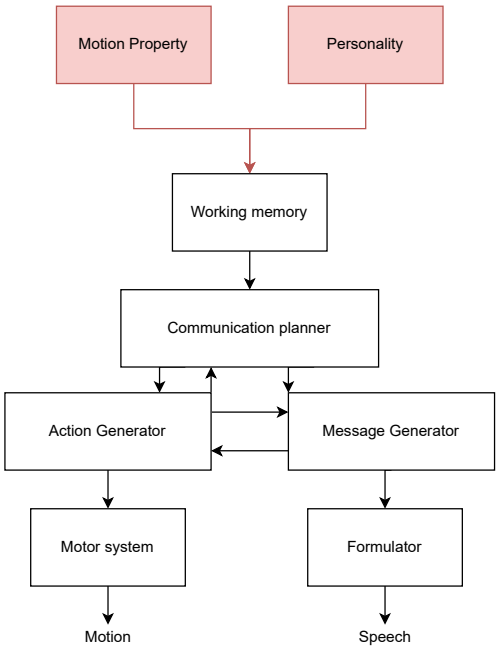
(a) Krauss and Hadar [1999]



(b) de Ruiter [2000]



(c) Kita and Özyürek [2003]



(d) Motion production with additional dependencies

Figure 1.1: Head motion production models. (d)is proposed by this thesis to illustrate additional dependencies shown in red

Chapter 2. However, direct waveform input has proved to be difficult to learn for automatic speech recognition (ASR) compared to hand-crafted features (Tüske et al. [2014]). We believe it would be even harder in head motion generation because waveform and motion are two data streams whereas speech and text in ASR are only one. In synthesising time, since the frequency of the speech is much faster than the frequency of the head movement (Hofer and Shimodaira [2007]), we usually apply upsampling methods in the head motion data. This resulting head motion coordinates react quicker, thus creating jerky or discontinuous. Furthermore, motions are always produced without awareness and they are impossible to be defined accurately as spontaneous (Riggio and Riggio [2012]). This leads another challenge in measuring the quality of the systems by just computing the numerical difference between the generated and original motions (Kucherenko et al. [2019]).

To make waveform suitable for head motion generation, we present methods that compress the waveform to be low dimensional and highly correlated with head motion, in addition to methods that enable the generated head motion to be more natural than the extracted feature. In the synthesis, we present a method to de-noise the generated motions for smoothness and a new objective measure to evaluate the systems in different aspects. For recurrent-neural-network-based (RNN-based) head motion models, we investigate the usefulness of the recurrent unit and aim to prove the effectiveness of our proposed feature in head motion generation based on our investigation.

During our investigation of the motion model training, we generalise our study to the training of deep neural networks (DNNs). We use multiple frames of speech features to map a frame of motion (Ding et al. [2015a];Haag and Shimodaira [2016];Kucherenko et al. [2019]). That is because a distinct head motion can last at least over 400ms (Hofer and Shimodaira [2007]), which is equivalent to 41 frames of speech features if 25ms windowing and 10ms shifting are applied in speech pre-processing. In such way, if we would like to generate multiple frames of motions, we either repeat the procedure

multiple times or input multiple blocks of frames of speech features at once. Either way is time-consuming and complicated. Then, these predicted frame-based motions are concatenated in the synthesis, this combined result is either lagging or jerky (Busso et al. [2007a]; Ding et al. [2015a]). Nevertheless, the temporal information of the generated motion is little considered in frame-based manner.

However, to the best of our knowledge, there are few works on building a motion generator in sequentially. Even though the recurrent unit or auto-regressive technique can carry information throughout the time domain, the input and output of the model are still frame by frame. Therefore, we present a frame-based method that can take input and generate output sequentially. We also aim to compare our model's results with other teams under a common dataset in Genea2020 Challenge Kucherenko et al. [2020].

1.3 Thesis Scope

This dissertation aims to provide a deep learning technique that directly maps speech waveform to motions. This research can benefit many applications directly or indirectly. For example, its findings can be applied in real life on conversation agents and further encompasses a wide range of applications from interaction with corporate websites to virtual psychologists, and currently embodying them (creating virtual avatars) is becoming more popular. Another goal of this thesis is to provide a set of evaluation measures to verify the quality of the generated motions. At the time of writing, no study has previously tried to use speech waveform directly to generate motion, and this remains an open research question.

In summary, the scope of this thesis is outlined as follows:

- **Motion in the conversation**: A large amount of motions are spontaneously

emitted during conversions and transmitted as non-verbal signals to the listeners; hence, substantial data can be collected, facilitating training.

- **Head/gesture/body motion**: These three types of motion occur the most in conversations and are the expression of the speaker/listener.

- **Mapping speech waveform to motion**: During conversations, there are two types of signals: verbal and non-verbal. We believe that there is a link between both signals and that we could use one of them to estimate the other.

- **Neural network**: Neural network is a type of parametric model, whose parameters are learned from data. This means that it is not necessary to construct a posterior distribution from a prior and a specific model for the likelihood such as a mixture. The posterior is learnt directly from data, which is more efficient and less model-dependent.

In other words, this thesis excludes the following:

- **Lip motion**: Lip motion is more correlated with facial expression rather than speech itself. Using speech features alone can only estimate standard lip-reading motions.

- **Semantics gestures**: Semantics gestures will not be directly synthesised as this would require knowing the intention of the speaker, which is not possible using the speech features alone.

- **Hidden Markov Model (HMM) & rule-based approach**: HMM is not investigated here as the task in this thesis is a many-to-many problem. It is because HMM is dependent and sensitive, relying heavily on the choice and quality of the features and parameters which may affect their performance. Speech signals are not independent of each other, continuous, or multi-dimensional. The same logic can be applied to the rule-based approach as well.

1.3.1 Research statement

What is a better technique for mapping speech waveform to motions and how can these generated motions be evaluated?

These two statements can be broken into the following multiple points.

- What is the suitable technique to compress waveform to be a useful feature for the motion synthesis?
- What is the suitable technique to map the proposed feature to motions?
- What is the suitable technique to post-filter the predicted motions?
- Prove the model's validity through both objective and subjective evaluation. This in turn requires:
  - Developing new measures as needed
  - Designing and conducting subjective tests.

1.3.2 Contribution

In this thesis, we develop speech waveform that has the potential to be a popular feature to estimate motion and achieves a better or comparable performance than other popular spectral features. For an end-to-end system, our contributions are the following:

- **Deep Canonical Correlation Constrained Auto-Encoder (DCCCAE)**. The proposed DCCCAE learns not only minimises the reconstruction error between the input and output but also maximises the correlation between the embedded features and the head motions. By doing so, the proposed DCCCAE can compress the waveform to a relatively low dimension and retain useful information for the downstream tasks.
- **A trainable post-filtering neural network**. The proposed post-filtering method

can learn the movement property of head motion, and this filter has proved to be effective in de-noising data involving different types of noise (dropout/Gaussian noise) with a lower value in objective evaluation compared to the common impulse filters.

- **Feature analysis that demonstrates the effectiveness of DCCCAE in retaining useful information**. To investigate if DCCCAE is useful, we compare our proposed feature ($\text{Wav}_{\text{CCCAE}}$) with common acoustic features in multiple speakers using local canonical correlation analysis (CCA), and illustrating with T-SNE. We observe a higher correlation in $\text{Wav}_{\text{CCCAE}}$ than other selected features, and $\text{Wav}_{\text{CCCAE}}$ shows a clear personal dependence and shares a common motion property among different speakers.

- **Experiments that demonstrate that $\text{Wav}_{\text{CCCAE}}$ is better/comparable in generating the head motions**. To explore whether $\text{Wav}_{\text{CCCAE}}$ is useful in generating head motion, we propose to build a regression model individually with $\text{Wav}_{\text{CCCAE}}$, MFCC and another extracted feature (Wav_{AE}) with standard auto-encoder (AE), which is only trained with construction error. The motion generated from $\text{Wav}_{\text{CCCAE}}$ is better than Wav_{AE} and comparable with MFCC in normalised mean square error (NMSE) and local CCA.

- **Term-weighted Value (TWV)**. This proposed evaluation is conducted during head motion synthesis as it can visualise the similarities and differences between the generated and ground truth whereas other common metrics (e.g. mean square error (MSE) and CCA) only provide results regarding either the similarities or the differences. In our reporting results, the head motion predicted from $\text{Wav}_{\text{CCCAE}}$ has lower or comparable TWV (better performance) than head motion predicted from other features. Moreover, the visualised peak angles from $\text{Wav}_{\text{CCCAE}}$ are closer to the ground truth.

For recurrent unit regression, we present the following contributions:

- **Experiments that demonstrate that the long-short term memory (LSTM) unit is useful**. To investigate the effectiveness of LSTM, we propose to replace some feedforward layers in the regression model with LSTM layers. When the feedforward layers in the regression model are replaced with LSTM layers, we notice that the performance of the models is boosted with lower NMSE and higher local CCA, and the LSTM models adapt better with $\text{Wav}_{\text{CCCAE}}$.

- **Subjective Evaluation**. We perform subjective evaluations in a crowd-sourcing platform in two regards: appropriateness (with MUSHRA-like evaluation) and model assessment (with A/B test). MUSHRA-like evaluation demonstrates that the participants deemed models with $\text{Wav}_{\text{CCCAE}}$ to be better than the other models. The A/B test further highlights that the LSTM models adapt better with $\text{Wav}_{\text{CCCAE}}$.

For gesture and body estimation, our contributions are as follows:

- **Double-DCCCAE**. The proposed double-DCCCAE learns to compress waveform stream and motion stream individually into fixed frame-based embeddings by minimising construction error and maximising the correlation with the objective features. The embedded feature pair is in a low dimension and high correlation with each other. Then the predicted embedded motion from the embedded waveform can recover back to motion stream.

- **Subjective Evaluation**. We submitted our results to GENEA2020 Challenge, and they collected a total of five teams over the world, plus two baseline models, one anchor created by them and the ground truth to perform subjective evaluation online in two aspects: human-likeness and appropriateness. Our model was more preferred than the baseline model trained with only audio in terms of human-likeness and comparable in terms of similar appropriateness.

1.4 Thesis Outline

Chapter 1: Introduction. This chapter introduces the key factors of the human-computer communications, the definition of communications, the production of motions and their links. It presents open issues that are related to speech-driven head motion. The problem statement and contributions of the thesis are discussed.

Chapter 2: Background. Relevant related work is presented and summarized.

Chapter 3: Speech-driven Head Motion System with Waveform. We present an end-to-end speech-driven system, which takes speech waveform directly, and estimate head motion. We demonstrate empirical evidence that the proposed feature achieves comparable state-of-the-arts (SOTA) results objectively. We also develop a new objective metric to evaluate the system. Based on the metric, we analyse the estimated head motion in occurrence.

Chapter 4: LSTM-based Head Motion Estimation with DCCCAE. We present a RNN-based regression method to further boost our proposed feature in estimating head motion. In our experiments, we show that the proposed feature is better than MFCC features in objective evaluations. We also perform a MUSHRA-like subjective evaluation. The participants chose the model with our proposed feature as the best over others, excluding the ground truth.

Chapter 5: Upper Body Motion Estimation Using Double-DCCCAE. We extend our approach to gesture and body motion estimation. In this investigation, we use two DCCCAE instead of only one in head motion estimation. In objective evaluation, we achieve comparable results compared to MFCC. In GENEA workshop's subjective evaluation, we outperform other approaches, which takes speech and text information together as input, in some respects.

Chapter 6: Conclusion. Final thoughts and considerations are discussed.

1.5 Publication

The thesis is based on the following published works:

- Lu, J., Shimodaira, H. (2020) Prediction of Head Motion from Speech Waveforms with a Canonical-Correlation-Constrained Autoencoder. Proc. Interspeech 2020, 1301-1305, doi: 10.21437/Interspeech.2020-1218
- Lu, J., Liu, T., Xu, S. and Shimodaira, H., 2021, June. Double-dcccae: Estimation of body gestures from speech waveform. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 900-904). IEEE.
- Lu, J. and Shimodaira, H., 2019. A neural network based post-filter for speech-driven head motion synthesis. Manuscript.

The following publications are not related to this thesis:

- Hu, S., Zhang, B., Lu, J., Jiang, Y., Wang, W., Kong, L., Zhao, W. and Jiang, T., 2022. WideResNet with Joint Representation Learning and Data Augmentation for Cover Song Identification. Proc. Interspeech 2022, pp.4187-4191.
- Yijun Zhou, JinHong Lu, Xiang Chen, Chia-Ming Chang, Takeo Igarashi. RelRoll: A Relative Elicitation Mechanism for Scoring Annotation with A Case Study on Speech Emotion. Graphics Interface 2023.

Chapter 2

Background

2.1 Introduction

Motion synthesis forms part of the larger field of character animation. The need for good non-verbal communication channels, such as head motion, cannot be understated for use in embodied characters. With the correct body language, an embodied conversational agent or any other animated character can appear, ”credible, trustworthy, confident, and non-threatening” (André et al. [2011]).

Current motion synthesis systems are mostly based on deep neural networks using text, audio and other information as the input. The motion generated from audio is also called a beat motion and the ratio of the beat motion is more than 70% of actual human motion (Mcneill [1994]). Thus, in other words, it is more than enough to generate most of the human motion from speech for the embodied characters to act as human likenesses. It is one of the key factors for us to investigate speech-driven motion synthesis systems.

In the rest of this chapter, we introduce a correlational neural network, which is the key architecture we will study thoroughly in this thesis. Moreover, the experiments in this

thesis require background knowledge in several key areas: key theoretical concepts in motion synthesis, motion regression system and the methodology of the evaluations. This chapter also provides such background for the reader and introduces important concepts that will be referenced in later chapters.

2.2 Representation Learning

Research on speech processing has traditionally considered the task of designing hand-crafted acoustic features (feature engineering) as a separate distinct problem from the task of designing efficient machine learning models to make prediction and classification decisions. There are two main drawbacks to this approach: first, the feature engineering being manual is cumbersome and requires human knowledge, and second, the designed features might not be best for the objective at hand (Latif et al. [2023]). This has motivated the adoption of a recent trend in the speech community towards utilisation of representation learning techniques, which can automatically learn an intermediate representation of the input signal that better suits the task at hand, hence leading to improved performance. The significance of representation learning has increased with advances in neural network, where the representations are more useful and less dependent on human knowledge, making it very conducive for tasks such as classification, prediction, etc. This section introduces the main architecture that will be used in this thesis: correlational neural network (CorrNet) (Chandar et al. [2016]), which is a type of representation learning.

Before delving into CorrNet, let us start with the development of representation learning. Learning representation aims to capture useful information or attributes of data, where deep representation learning involves neural network models for this task. Various applications of deep representation learning have been summarised in Latif et al. [2023]:

- Feature learning: a process of constructing explanatory variables or features that can be used for classification and prediction problems.
- Abstraction and invariance: a process of discovering a universal model that can be applied across all tasks to facilitate generalisation and knowledge transfer.
- Disentanglement and manifold learning: a method that disentangles or represents each feature into narrowly defined variables and encodes them as separate dimensions.
- Clustering structure: a process of categorising similar classes of data samples into one cluster using similarity measures.
- Data de-noising: a process of filtering the noise of the data.
- Dimension reduction and information retrieval: a process of eliminating data redundancy and irrelevancy for higher efficiency, often increasing performance and making the data more understandable and interpretable by reducing the number of input variables.

In speech-related tasks, representation learning has been explored well. For text-to-speech (TTS), representation learning is performed to train a speaker 'code' to assist the system training to learn that some of the training data belong to different speakers as in the work of Luong et al. [2017]. Luong et al. [2017] found that one-hot speaker encodings resulted in better naturalness, but the discriminant condition codes resulted in higher speaker similarity. The discriminant condition codes modelled speaker characteristics in a way that was not possible with one-hot vectors because the representations were richer and accounted for more nuanced variation. The work of Jia et al. [2018] expanded on the idea of using speaker embedding in TTS. They introduced a TTS system based on three separately trained modules: a speaker encoder network, a synthesis network and a vocoder network. The purpose of the speaker encoder is to learn to encode speaker identity from waveform input into an embedding that can

be used in the synthesis network. Their experiments show that the speaker embeddings helped achieve high-quality naturalness and speaker similarity on both datasets. Then for ASR, representation learning aims to learn discriminative and robust representation. Hsu et al. [2017] used FHVAE to capture interpretable and disentangled representations from speech without any supervision. They evaluated the model on two speech corpora and demonstrated that FHVAE can satisfactorily extract linguistic contents from speech and outperform an i-vector baseline speaker verification task while reducing word error rate (WER) for ASR. Sailor and Patil [2016] used a convolutional RBM (ConvoRBM) to learn auditory-like sub-band filters from the raw speech signal. The authors showed that unsupervised deep auditory freatures learnt by ConvoRBM can outperform Mel filterbank features.

In this thesis, we mainly focus on **dimension reduction and information retrieval** because we aim to compress the speech waveform into a low dimension and extract information, which high correlation with motion from speech waveform, resulting a better estimation in later stage. It has been validated that the use of more interpretable features in a lower dimension can provide compretitive performance or even better performance when used in designing predictive models (Latif et al. [2018]). There are two popular paradigms: Canonical Correlation Analysis (CCA) and approaches based on Autoencoder (AE). AE-based methods learn a representation by minimizing the error of reconstructing the views, and this will be discussed in the later section. CCA-based approaches learn a joint representation by maximizing correlation of the views when projected to the subspaces.

2.2.1 CCA-Based Approaches

CCA was first introduced by (Hotelling [1936]; Anderson [2009]) to find linear projections of two random vectors that are maximally correlated in standard statistics. CCA is useful in learning representations of two data views such that each view's rep-

resentation is simultaneously the most predictive of, and the most predictable by, the other (Andrew et al. [2013]). An appealing property of CCA for prediction tasks is that if there is noise in either view that is uncorrelated with the other view, the learned representations should not contain the noise in the uncorrelated dimensions (Andrew et al. [2013]). The appealing property makes CCA suitable in wide ranges of tasks and fields, including performing unsupervised data analysis when multiple views are available (Hardoon et al. [2007]; Vinokourov et al. [2002]), learning features for multiple modalities that are then fused for prediction (Sargin et al. [2007]), learning features for a single view when another view is available for representation learning but not at prediction time (Blaschko and Lampert [2008]; Chaudhuri et al. [2009]), and reducing sample complexity of prediction problems using unlabeled data (Kakade and Foster [2007]).

CCA is defined by the following equation, which finds pairs of linear projections of the two views that are maximally correlated:

$$(\mathbf{w}_1^*, \mathbf{w}_2^*) = \underset{w_1, w_2}{\arg\max}\, \text{corr}(\mathbf{w}_1^{'}\mathbf{X}_1, \mathbf{w}_2^{'}\mathbf{X}_2) \tag{2.1}$$

$$= \underset{w_1, w_2}{\arg\max} \frac{\mathbf{w}_1^{'}\Sigma_{12}\mathbf{w}_2}{\sqrt{\mathbf{w}_1^{'}\Sigma_{11}\mathbf{w}_1\mathbf{w}_2^{'}\Sigma_{22}\mathbf{w}_2}} \tag{2.2}$$

Let $\mathbf{w}_1$ is the $n_1 \times 1$ vector, $\mathbf{w}_2$ is the $n_2 \times 1$ vector, $(\mathbf{X}_1, \mathbf{X}_2) \in \mathbb{R}^{n_1 \times T} \times \mathbb{R}^{n_2 \times T}$ denote random vectors with covariances $(\Sigma_{11}, \Sigma_{22})$ and cross-covariance Σ_{12}, linear projections of two views $(\mathbf{w}_1^{'}\mathbf{X}_1, \mathbf{w}_2^{'}\mathbf{X}_2)$.

Since the objective is invariant to scaling of w_1 and w_2, the projections are constrained to have unit variance:

$$(\mathbf{w}_1^*, \mathbf{w}_2^*) = \underset{w_1^{'}\Sigma_{11}w_1 = w_2^{'}\Sigma_{22}w_2 = 1}{\arg\max} \mathbf{w}_1^{'}\Sigma_{12}\mathbf{w}_2 \tag{2.3}$$

When finding multiple pairs of vectors $(\mathbf{w}_1^i, \mathbf{w}_2^i)$, subsequent projections are also con-

strained to be uncorrelated with previous ones, that is, $\mathbf{w}_1^i \Sigma_{11} \mathbf{w}_1^j = \mathbf{w}_2^i \Sigma_{22} \mathbf{w}_2^j = 0$ for $i < j$. Assembling the top k projection vectors $\mathbf{w}_1^i$ into the columns of a matrix $\mathbf{A}_1 \in \mathbb{R}^{n_1 \times k}$ and similarly placing $\mathbf{w}_2^i$ into $\mathbf{A}_2 \in \mathbb{R}^{n_2 \times k}$, we obtain the following formulation to identify the top $k \leq \min(n_1, n_2)$ projections:

$$maximize : tr(\mathbf{A}_1^{'} \Sigma_{12} \mathbf{A}_2) \tag{2.4}$$

$$\text{subject to} : \mathbf{A}_1^{'} \Sigma_{11} \mathbf{A}_1 = \mathbf{A}_2^{'} \Sigma_{22} \mathbf{A}_2 = I. \tag{2.5}$$

There are several ways to express the solution to this objective; we follow the one in (Mardia et al. [1979]). Define $T \triangle \Sigma_{11}^{-1/2} \Sigma_{12} \Sigma_{22}^{-1/2}$, and let U_k and V_k be the matrices of the first k left- and right- singular vectors of T. Then the optimal objective values is the sum of the top k singular values of T (the Ky Fan k-norm of T), and the optimum is attained at $(\mathbf{A}_1^*, \mathbf{A}_2^*) = (\Sigma_{11}^{-1/2} U_k, \Sigma_{22}^{-1/2} V_k)$. Note that this solution assumes that the covariance matrices Σ_{11} and Σ_{22} are nonsingular, which is satisfied in practice because they are estimated from data with regularization: given centered data matrices $\bar{H}_1 \in \mathbb{R}^{n_1 \times m}, \bar{H}_2 \in \mathbb{R}^{n_2 \times m}$, one can estimate, e.g.:

$$\widehat{\Sigma_{11}} = \frac{1}{m-1} \bar{\mathbf{H}}_1 \bar{\mathbf{H}}_1^{'} + r_1 \mathbf{I}, \tag{2.6}$$

where $r_1 > 0$ is a regularization parameter. Estimating the covariance matrices with regularization also reduces the detection of spurious correlations in the training data, which is also known as ”overfitting” (De Bie & De Moor, 2003).

Deep CCA is the first attempt of the investigations on CCA and deep learning, and was introduced by (Andrew et al. [2013]). Deep CCA computes representations of two views by passing them through multiple stacked layers of nonlinear transformation

(see Figure 2.1). The output of the views can be defined by the following formula:

$$\mathbf{v}_1 = f_1(\mathbf{X}_1;\theta_1) \tag{2.7}$$

$$= s(\mathbf{W}_l^1 s(\mathbf{W}_1^1\mathbf{X}_1 + \mathbf{b}_1^1) + \mathbf{b}_l^1) \text{ for } 2 <= l < d \tag{2.8}$$

$$\mathbf{v}_2 = f_2(\mathbf{X}_2;\theta_2) \tag{2.9}$$

$$= s(\mathbf{W}_l^2 s(\mathbf{W}_1^2\mathbf{X}_2 + \mathbf{b}_1^2) + \mathbf{b}_l^2) \text{ for } 2 <= l < d \tag{2.10}$$

where θ_1 is the vector of all parameter W_l^1, θ_2 is the vector of all parameter W_l^2, $W_l^1 \in \mathbb{R}^{c_1 \times n_1}$ is a matrix of weights, $b_l^1 \in \mathbb{R}^{c_1}$ is a vector of biases, $W_l^2 \in \mathbb{R}^{c_2 \times n_2}$ is a matrix of weights, $b_l^2 \in \mathbb{R}^{c_2}$ is a vector of biases, and $s : \mathbb{R} \to \mathbb{R}$ is a nonlinear function applied componentwise. The goal of this model is to jointly learn parameters for both views $W_l^v and b_l^v$ such that $corr(f_1(X_1), f_2(X_2))$ is as high as possible.

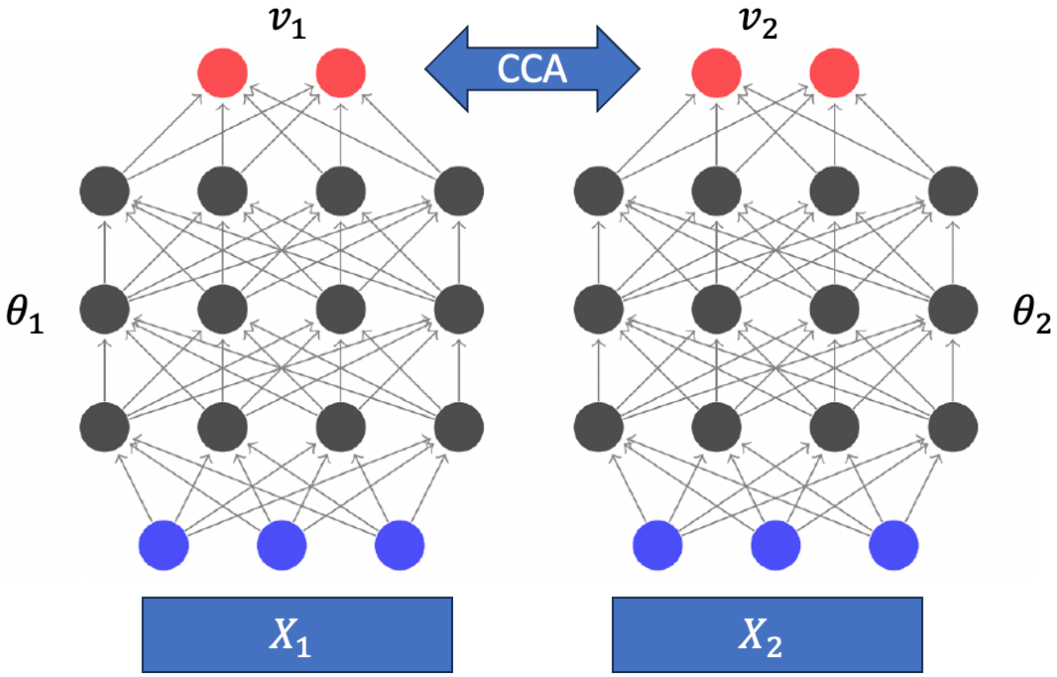


Figure 2.1: Deep CCA model architecture Andrew et al. [2013]. X_1 and X_2 are two different data views and v_1 and v_2 are the representation of the two corresponding data views after the two non-linear transformations θ_1 and θ_2.

2.2.2 AE-Based Approaches

The AE-based methods mentioned in the above section aim to minimise the reconstruction error resulting in low dimensional representation. Multimodal auto-encoders (MAEs), an example of AE-based methods, have been proposed to learn a common representation for two views/modalities (Ngiam et al. [2011]). The idea in MAE is to train an AE to perform two kinds of reconstruction (shown in Figure 2.2).

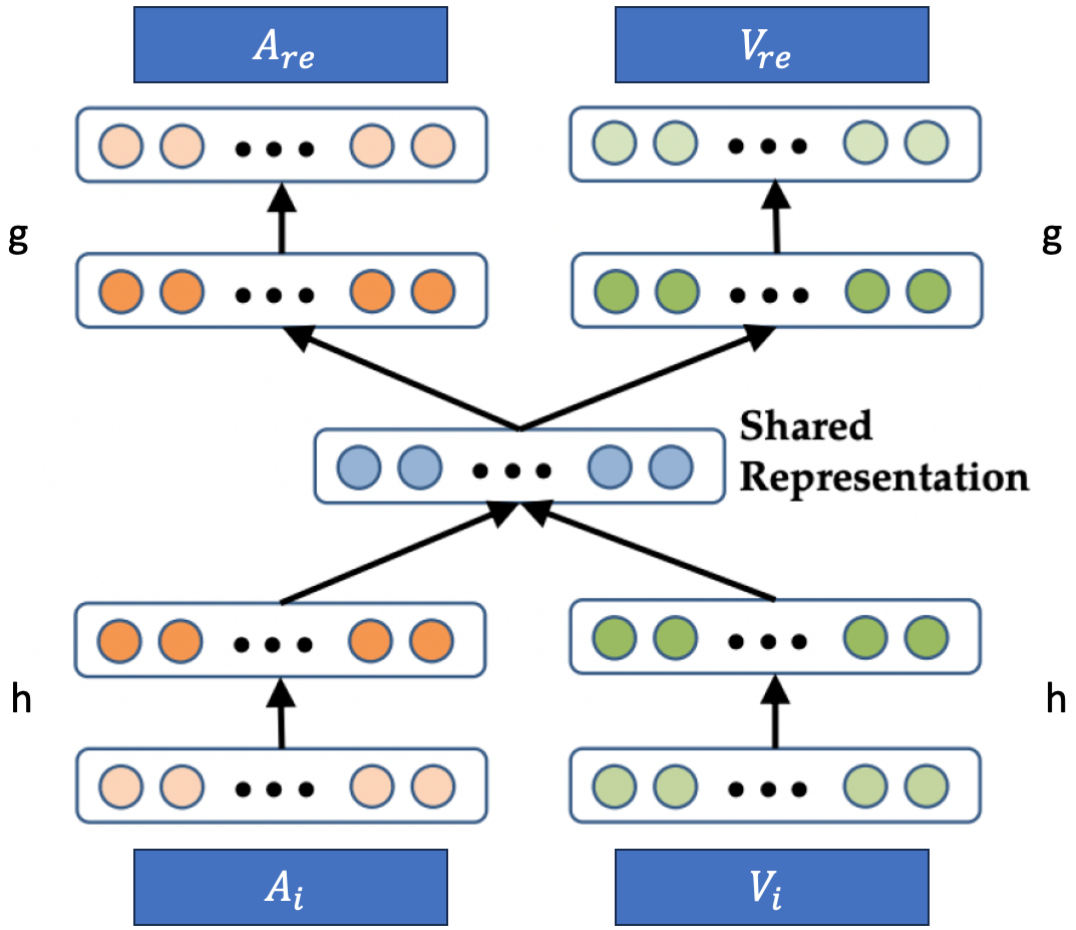


Figure 2.2: MAEs architecture. A_i and A_{re} are the audio input and reconstructed audio, V_i and V_{re} are the video input and reconstructed video. h and g are the encoder and decoder respectively.

Given any one view, the model learns both self-reconstruction and cross-reconstruction

(reconstruction of the other view). The objective function is shown as follows:

$$\mathcal{J}(\theta) = \sum_{i=1}^{N} (L(A_i, g(h(A_i))) + L(V_i, g(h(V_i))) + L(A_i, g(h(V_i))) + L(V_i, g(h(A_i))) \tag{2.11}$$

where the A and V are the audio and video input, function h and g are the encoding and decoding blocks, L is the reconstruction error and θ are the parameters in encoding and decoding blocks.

This makes the representation learnt to be predictive of the two views. However, MAE does not get any explicit learning signal encouraging it to share the capacity of its common hidden layer between the views, since the views are not guaranteed to be projected to a common subspace. The results reported in (Ngiam et al. [2011]) demonstrate that CCA performs better than deep MAE for the task of transfer learning.

2.2.3 CorrNet

With the definition of CCA, we note that it aims to produce correlated common representations but does not guarantee low dimension. Thus, CorrNet is proposed by (Chandar et al. [2016]) with the usage of MAEs and CCA. CorrNet is a method for learning representations, and it consists the following advantages:

- Allows for self/cross reconstruction
- The learnt representations are correlated to the target view
- Scalable when dealing with substantial high-dimensional data
- Easily modified to benefit from additional single-view data

The architecture of CorrNet is illustrated in Figure 2.3 and contains three layers: an input layer, a hidden layer and an output layer. Given $z = (x, y)$, the hidden layer

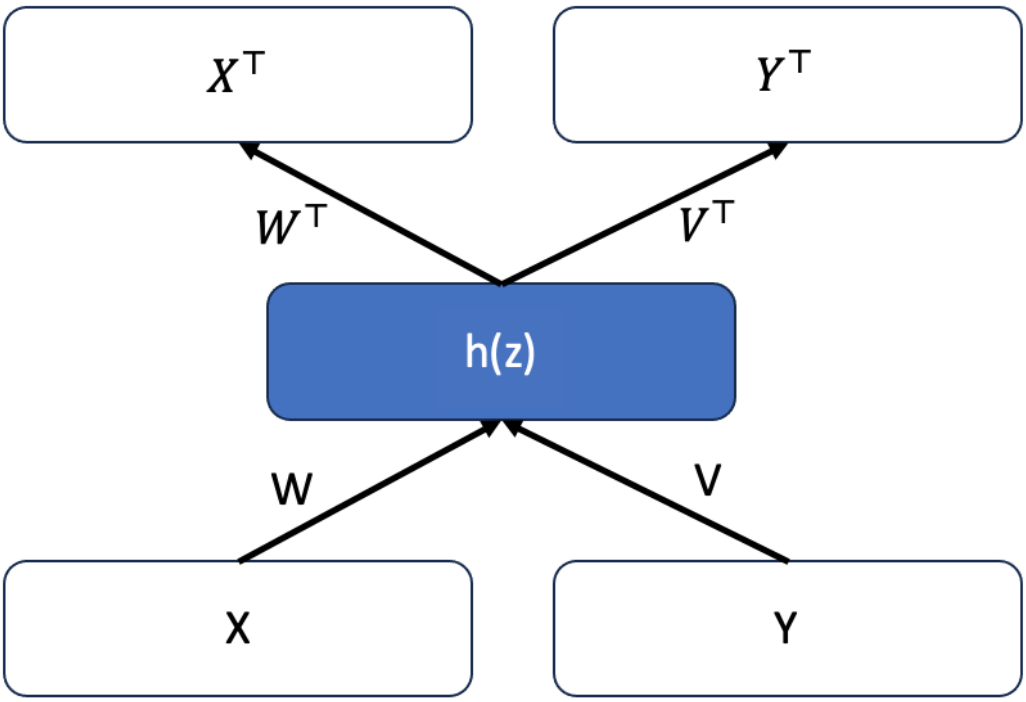


Figure 2.3: CorrNet architecture Chandar et al. [2016]. X and Y are the two data views, $X^\top$ and $Y^\top$ are the two reconstructed data views. W and V are the parameters of the encoder and $W^\top$ and $V^\top$ are the parameters of the decoder.

computes an encoded representation as follows:

$$h(z) = f(\mathbf{W}x + \mathbf{V}y + \mathbf{b}) \tag{2.12}$$

where x and y are the two view inputs, the $\mathbf{W}$ is a $k \times d_1$ projection matrix, $\mathbf{V}$ is a $k \times d_2$ projection matrix and $\mathbf{b}$ is a $k \times 1$ bias vector. Function f can be any non-linear activation function. The output layer then tries to reconstruct $\mathbf{z}$ from this hidden representation by computing:

$$\mathbf{z}' = g([\mathbf{W}'h(\mathbf{z}), \mathbf{V}'h(\mathbf{z})] + \mathbf{b}') \tag{2.13}$$

where $\mathbf{W}'$ is a $d_1 \times k$ reconstruction matrix, $\mathbf{V}'$ is a $d_2 \times k$ reconstruction matrix and $\mathbf{b}'$ is a $(d_1 + d_2) \times 1$ output bias vector. Vector $\mathbf{z}'$ is the reconstruction of $\mathbf{z}$. Function g can be any activation function. Thus, the parameters of the model are $\theta =$

$\{\mathbf{W}, \mathbf{V}, \mathbf{W}^{'}, \mathbf{V}^{'}, \mathbf{b}, \mathbf{b}^{'}\}$ and are optimised by the following objective function:

$$\mathcal{J}_Z(\theta) = \sum_{i=1}^{N} (L(\mathbf{z}_i, g(h(\mathbf{z}_i))) + L(\mathbf{z}_i, g(h(x_i))) + L(\mathbf{z}_i, g(h(y_i))) - \lambda \text{corr}(h(\mathbf{X}), h(\mathbf{Y})) \tag{2.14}$$

$$\text{corr}(h(\mathbf{X}), h(\mathbf{Y})) = \frac{\sum_{i=1}^{N} (h(x_i) - \overline{h(\mathbf{X})})(h(y_i) - \overline{h(\mathbf{Y})}))}{\sqrt{\sum_{i=1}^{N} (h(x_i) - \overline{h(\mathbf{X})})^2 \sum_{i=1}^{N} (h(y_i) - \overline{h(\mathbf{Y})})^2}} \tag{2.15}$$

where L is the reconstruction error, λ is the scaling parameter to scale the fourth term with respect to the remaining three terms, $\overline{h(\mathbf{X})}$ is the mean vector for the hidden representations of the first view and $\overline{h(\mathbf{Y})}$ is the mean vector for the hidden representations of the second view.

2.3 Key Theoretical Concepts

In the following section, we present some of the most important techniques and concepts at a theoretical level. First, we will explain some of the speech features that are available, such as F0, that can be the inputs to the head motion system. Moving on, we will describe some details about how rotation can be represented and explain why we chose to use rotation vectors. Finally, we will provide a review of the correlation between speech and head motion.

2.3.1 Speech Features

In ASR and TTS systems, several methods are widely used to encode speech. This is to make it easier to build statistical models compared to modelling the original waveform. Some examples of encoding systems are mel-frequency cepstral coefficients (MFCC), the related mel-cepstral coefficients (MCEP) and their generalised form mel-generalized cepstral coefficients (MGCEP). These are all based on representing the

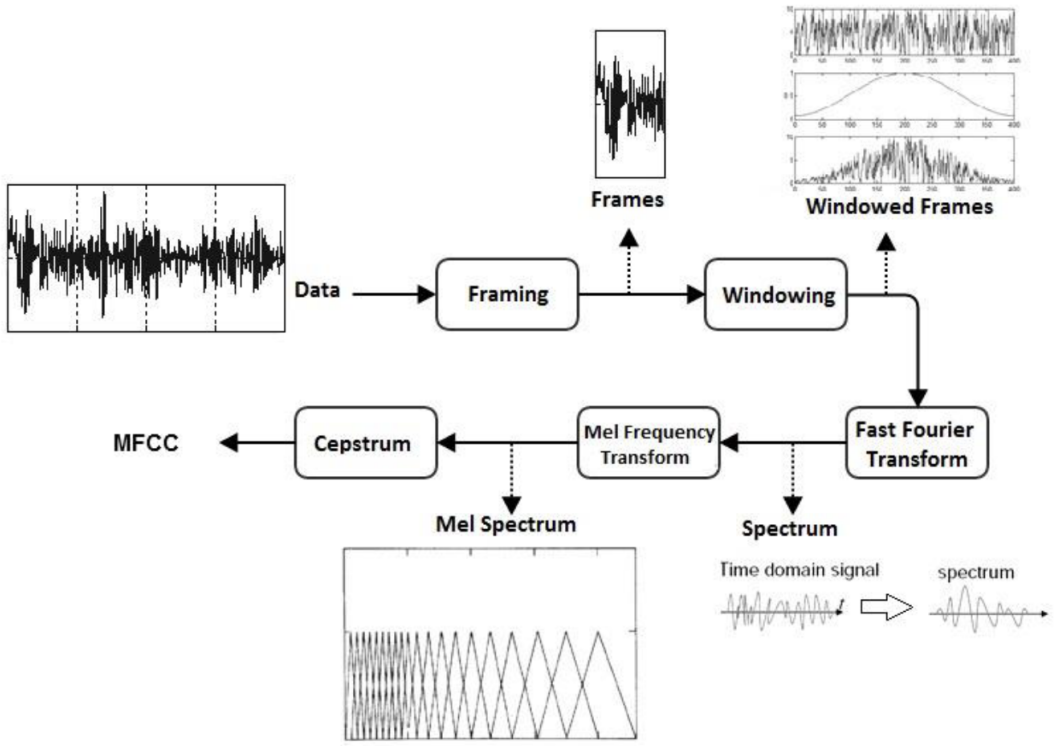


Figure 2.4: The flow procedure of the MFCC feature extraction (Arslan and Yildiz [2018])

cepstrum: the power spectrum of a sound. Normally, for speech, the cepstrum is determined over a window in time that moves at a fixed rate through the signal. For example, in TTS applications, a common choice of parameters is 5 to 10 ms frame shift with a 25 ms window over which the analysis is performed to extract the speech features. The procedure is illustrated in Figure 2.4.

A simple explanation of how waveforms are converted into the various coefficients is that a bank of lifters (cepstral domain filters) are applied to the cepstrum of the waveform by Fourier Transform, and the energy of the signal in each of the lifters forms the mel-scaled coefficients by triangular overlapping windows. The concept of the cepstrum was introduced by Bogert [1963] and it serves as a tool to investigate periodic structures in frequency spectra. The various types of mel-coefficients are different ways of constructing the signals and are designed to approximate the parts of the signal, which are inspired by human perception or hearing (Oxenham [2018]). For

humans, the perception of hearing is between 20 Hz and 20000 Hz (as per classical textbooks) and 80 Hz and 8000 Hz (as per psychophysics) give rise to auditory sensation (Poeppel and Teng [2020]). It is thought that these bands are the most important for comprehension of speech, both for understanding what was said by a human and for making synthesised speech understood.

A related set of features to the MFCC, MCEP and MGCEPs are linear predictive Coefficients (LPC) and linear spectral pairs (LSP). With the correct parameters, the MGCEPs are equal to LPCs, and there are algorithmic conversions between LSPs and LPCs. LPCs and LSPs are another way of dividing up the cepstrum while focusing on the ease of signal processing and interpolation, and not human hearing characteristics.

Another category of speech features are often called prosodic features. It is important here to differentiate between perceptual features and measurable features. As the name implies, perceptual features are what a listener would be able to perceive. Examples of perceptual features are pitch and loudness. On the other hand, measurable features can either be approximated or directly measured from a speech signal. While pitch is perceptual, the fundamental frequency of the glottal folds (sometimes called vocal chords) can be approximated, and this is called F0. While loudness is perceptual, we can measure the energy of the signal over the window. F0 and energy can function as stand-ins for pitch and loudness.

A significant problem with F0 is that it is not continuous in time. The glottal folds do not always vibrate during speech: for instance, when pronouncing the English letter 's', there is no movement in the glottis; instead, the sound is created by the shape of the lips, tongue position and air from the lungs. The region of the signal where F0 exists is called the voiced region, and the area where it does not exist is called the unvoiced region. There are two common methods for dealing with this problem. On one hand, when building the models, the F0 can be handled differently than the other features,

considering that it is not always measurable, Alternatively, F0 can be interpolated in the silence regions, although this would be an approximation.

Many software tools are available for extracting all of the above speech features from speech signals. Some common ones are SPTK, openSmile(Eyben et al. [2010], and STRAIGHT(Kawahara [2006]). Some of these tools also include methods for estimating the perceptual features from the speech features. For instance, openSmile outputs a pitch feature.

Another method for representing speech is based around how speech is produced in humans. In humans, speech is produced by the movement of air over the lips, tongue, teeth, and flottis (also known as the voice box). These are collectively known as the articulators. Studies on human speech production have measured how the articulators move during speech, initially using x-rays, and later (once the dangers of x-rays became known) employing a device known as the electro-magnetic articulograph (EMA). An EMA machine has the disadvantage of not being able to measure the glottal movements directly but, compared to x-rays, has the advantage of not giving study participants cancer. An EMA works by attaching magnetic coils to the articulators of the participant other than the glottis and determining the coils' movement by measuring changes in the magnetic field. Often, only two dimensions are considered. This is because in most languages, during non-impaired speech, the articulator movement is symmetrical about the left-right axis when facing the speaker.

Through a process known as speech inversion, EMA measurements, i.e., the movement of articulators, can be estimated using speech features. Recent research has examined the use of EMA features estimated from speech for head motion synthesis (Youssef et al. [2013];Ben Youssef et al. [2014]). It was found that predicted EMA features are more highly correlated with head motion than the standard array of speech features used in ASR and TTS.

2.3.2 Representing Rotation

Head motion can be described in rotation, and there are many representations for rotation itself. The most common ones are using three angles (DAVENPORT [1973]), quaternions (Shoemake [1985]), axis-angle or rotation vectors (Curtright et al. [2014]), and rotation matrix (Wigner [2012]), shown in Figure 2.5. The three angles are also known as, Euler Angles. However, a number of problems arise when using Euler angles, shown in Figure 2.5(a). The first is that there are singularities (Mortari et al. [2000], Curtis [2014]); in this regard, the actual rotation of the object is ambiguous, but this does not affect head motion synthesis as this condition only happens at the poles, and the normal range of human head motion is not that large. A far greater problem is that they are order dependant (Ohkami [2003]). By this we mean that applying the rotations in the order $\alpha\,\beta\,\gamma$ is not equivalent to the order $\gamma\,\alpha\,\beta$. At first glance, this may not seem to be an issue; however, in the literature, the order used is often not included when reporting results. While working on one's own programs, it is trivial to be consistent, and when collaborating with other researchers, trying to reproduce results in the literature or using commercial software,the order may not be obvious. Another problem is that the axes of rotations are not fixed (Ohkami [2003]). It is possible to represent any 3D rotation using any successively orthogonal axis; for instance, rotation about the y-axis, then x-axis and then y-axis again can represent any rotation. Another problem is that where the head is facing is also not known unless reported, but this ambiguity is common among rotation representation methods. To address this, a common convention in the literature is to use 'yaw', 'roll' and 'pitch'(Curtis [2014]); however, this still does not satisfy the order ambiguity.

The use of a rotation matrix does solve many of the issues of Euler angles. There is no order dependence because the rotation matrix rotates the object simultaneously about all axes Poznyak [2021]. It is also possible to convert a rotation matrix to a giver order of Euler angles, and if the object is not at a singularity, it is also possible to convert

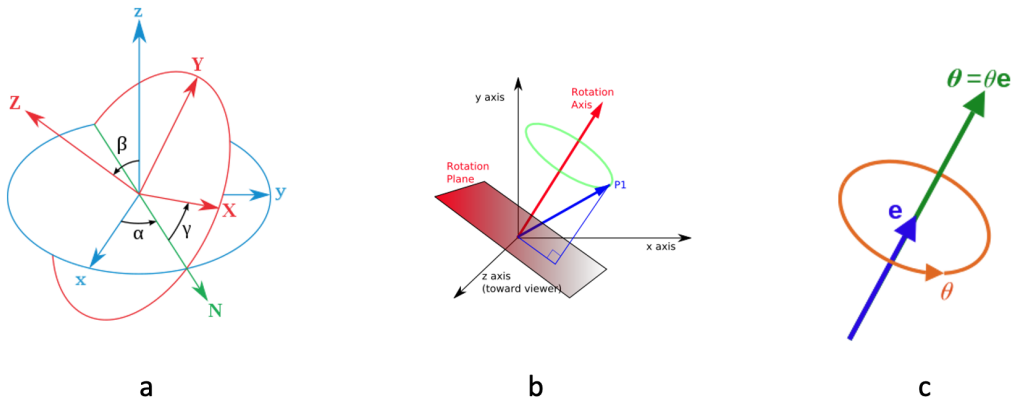


Figure 2.5: (a): Three angles (or called Euler angles). The blue lines (x, y, z) are the fixed coordinate system, the red lines (X, Y, Z) are the rotated coordinate system and the green lines are the nodes. (b): Quaternions. The scalar value, w, corresponds to an angle of rotation. The vector term, (x y z), corresponds to an axis of rotation, about which the angle or rotation is performed. (c): axis-angle representation of rotation. The angle θ and axis unit vector e define a rotation, concisely represented by the rotation vector θe

from Euler angles to a rotation matrix(Poznyak [2021]). Although each of the possible orders has its own conversion.

However, there are several ambiguities in rotation matrix. The first is that right- or left-handed coordinatescan result in the vector being represented differently with respect to a different coordinate system (Flórez-Valencia and Orkisz [2017]). Another ambiguity is that pre-multiplication or post-multiplication, which refers to a same point P, can be represented either by a column vector v or a row vector w. Rotation matrices can either pre-multiply column vectors (Rv) or post-multiply row vectors (wR). However, Rv produces a rotation in the opposite direction with respect to wR(Poznyak [2021]). Moreover, it is difficult to interpret the meaning of individual elements of the rotation matrix. The rotation matrix consists of nine elements for three dimensional rotation. The redundancy makes it difficult to visualise the rotation represented by the matrix without actually using a computer (Razavian et al. [2019]). In other words, there is no natural meaning to any one of the elements of the matrix. If we were to try and synthesize a rotation matrix directly, it must satisfy the following constraints:

- Orthogonal
- Determinant of 1
- Real entries

It should also be noted that the addition of multiple rotations is done through matrix multiplication. If the object is rotated by matrix R_1 and then R_2, the total rotation R_T is given as follows:

$$R_T = R_2 R_1 \tag{2.16}$$

While this is certainly not problematic, it would mean that calculating the differentials of the rotation for the angular velocity and acceleration would be more difficult.

The key difference between a rotation matrix and Euler angles is the amount of parameters. The addition of six extra parameters is responsible for the lack of ambiguity, but there is redundancy in this information; hence, the resulting co-dependence of the elements of a rotation matrix make synthesis difficult (Razavian et al. [2019]). On the other hand, Euler angles are independent in terms of rotation, thus making synthesis easier. The independence we are referring to is mathematical independence; in this regard, it is highly likely that when representing the movement of the head Euler angles, cross dependencies would be encountered (Ohkami [2003]).

Quaternions (Shoemake [1985]) have four parameters, shown in Figure 2.5(b). Quaternions similar to a rotation matrix. Quaternions describe rotation unambiguously. Thus, we can conclude that this is probably the ideal number of parameters. There is, however, still an issue with interpretation. Euler's rotation theorem states that any rotation in 3D can be represented by an axis about which the object will rotate and an angle that is the magnitude of the rotation. The four elements of a quaternion that represents rotation are by convention called: w, x, y and z.

This may lead one to believe that x, y and z are the axes of rotation and that w is the an-

gle (Jones [2004]). However, this is not the case. Quaternions have been formulated in such a way that applying multiple rotations is simple, but this means that the elements still have no easy interpretation.

The axis-angle representation of rotation (Curtright et al. [2014]) is a far more direct interpretation of Euler's rotation theorem, shown in Figure 2.5(c). As the name implies, the rotation is specified by a 3D vector, which is the axis of rotation, and an angle, which is the magnitude of the rotation. An extension of this representation is a rotation vector. A rotation vector is a conversion of the 4D axis-angle down to a 3D rotation vector. This is achieved by normalising the magnitude of the vector representation axis of rotation and then setting the magnitude of the rotation vector to be the angle of rotation. This replaces the forth parameter that we need to uniquely describe rotation with prior knowledge. Mathematically, if α is the angle and v is the axis of rotation, then the rotation vector r is given as follows:

$$r = \alpha \frac{v}{||v||} \tag{2.17}$$

The rotation vector (Curtright et al. [2014]) representation has many advantages. First, it is unambiguous regarding order as one can think of it as simultaneous rotation about all three axes. Second, the components are measured in radians, so they are easy to interpret. Third, there are no singularities when there is a rotation, and no rotation is given by the zero vector, which is intuitive. The major disadvantage of rotation vectors is that they are not easy to add together. Normally, addition of rotations is done by converting to either quaternions or rotation matrices first (Diebel [2006]). What could also be a theoretical problem with this formulation is that there is a discontinuity at π radians; however, as the human head cannot turn this far, this problem can be disregarded for this application.

For the purposes of this research, we have chosen to use rotation vectors, for which the

component will be denoted as r_x, r_y and r_z.

The issue of addition is not problematic because we only specify absolute angles of rotation, and when applying the rotation, we will convert to a rotation matrix with:

$$\alpha = ||r|| \quad s = \sin\left(\frac{\alpha}{2}\right) \quad c = \cos\left(\frac{\alpha}{2}\right) \tag{2.18}$$

$$R = \begin{bmatrix} V_1(r) & V_2(r) & V_3(r) \end{bmatrix} \tag{2.19}$$

$$V_1(r) = \frac{1}{\alpha^2}\begin{bmatrix} (r_x^2 - r_y^2 - r_z^2)s^2 + \alpha^2 c^2 \\ 2s(r_x r_y s - \alpha r_z c) \\ 2s(r_x r_z s + \alpha r_y c) \end{bmatrix} \tag{2.20}$$

$$V_2(r) = \frac{1}{\alpha^2}\begin{bmatrix} 2s(r_x r_y s + \alpha r_z c) \\ (r_y^2 - r_z^2 - r_x^2)s^2 + \alpha^2 c^2 \\ 2s(r_y r_z s + \alpha r_x c) \end{bmatrix} \tag{2.21}$$

$$V_3(r) = \frac{1}{\alpha^2}\begin{bmatrix} 2s(r_x r_z s - \alpha r_y c) \\ 2s(r_y r_z s + \alpha r_x c) \\ (r_z^2 - r_x^2 - r_y^2)s^2 + \alpha^2 c^2 \end{bmatrix} \tag{2.22}$$

where R is the rotation matrix.

Finally, the rotation vector is the representation used in this research. Unless it is specified otherwise, the reader can assume we used rotation vectors throughout this thesis.

It is helpful to know that our model estimate, the head motions in the rotation of three trajectories (X, Y, Z) as shown in Figure 2.6, and the visualizing head motion soft-

ware is provided by my project supervisor, Dr Hiroshi Shimodaira. The input of this software is the rotation of XYZ in radian.

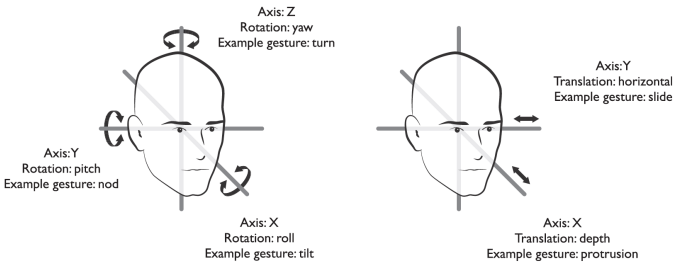


Figure 2.6: Schematic overview of rotations and translations along three axes, as well as example movements most frequently used in communicative head gesturing (Wagner et al. [2014])

2.3.3 Correlation between speech and head motion

Communication, regardless of its mode, is deemed an essential part of the existing civilization, and it consists of verbal and nonverbal forms. Hadar et al. [1983] argue that one important nonverbal form, head motion, directly contributes to speech production. Research on head motion is undergoing significant changes as the synthesis of head motion moves towards fully operational and interactive implementation. Such niche technology is being tested to apply both head motion and lip-syncing to the creation of a more human-like avatar. However, compared with lip-syncing, we may not reach the point at which head motions could be easily captured and analyzed due to a weak link between speech and head motion and a complex collective of speech, emotion, intention, and stance.

One of the earliest studies concerning the correlation between prosody and gesture was made by Birdwhistell [1952]. It was suggested that there is an alignment between gestural movements and intonation. Bolinger [1983] and Bolinger and Bolinger [1986] observed that gestures followed pitch contours up and down, in their main direction of movement. Hadar et al. [1983] also demonstrated that speakers' head movements

move along with the changes of the prosody, which are in peaks and their boundaries, especially in cases of high intensity.

In recent years, further analysis was made. Kuratate et al. [1999] found that fundamental frequency (F0) and head motion that represented in 'pitch-roll-yaw' (or called rotation vector) had a correlation of 0.83 at sentence-level, but they also claimed that this analysis was sensitive to the absolute values rather than the spatiotemporal patterning of head posture. Yehia et al. [2002] analysed the correlation between head motion and speech over the fundamental frequency (F0) by the experiment was conducted on the reading speech utterances of the ES and JS. They ascertained that the correlation among the F0 and the six DOF (degrees of freedom) (three DOF for rotation and three DOF for translation) of head motion was between 0.39 and 0.52 for ES and between 0.22 and 0.30 for JS, which are on average less than 0.50. Munhall et al. [2004] reported that the correlations between head motion (in six DOF) and pitch and amplitude of the speaker's voice were almost always over 0.50, on average about 0.63 in sentence-level, in Japanese read speech utterances. Busso and Narayanan [2007] presented an audio-visual mapping framework, which maps the acoustic features onto the facial features space, producing the estimated facial features by affine minimum mean square error estimator (AMMSE). These estimated facial features were then used to compute the Pearson's correlation with the real facial features. They showed a strong sentence-level correlation (r = 0.8) between head motion and MFCCs after the mapping, where data was recorded for an actor reading the scripts of short sentences.

Overall, the above studies have reported high correlation between scripted-speech and head motion. However, as we show in experiments, it is a different scenario in natural conversations, where there is a much larger degree of variation in head motion, and we cannot find such strong correlations. There are other studies to support our hypothesis as well. Ishi et al. [2007] analyzed spontaneous dialogue speech data from one Japanese female speaker and claimed that a strong relationship could not be found

between head motion and prosodic features. Sadoughi and Busso [2019] reported that a global CCA between the original head movements and speech (F0 and energy with their first and second order derivatives) is 0.1931 with the dyadic interactions data.

2.4 Motion Synthesis System

This section discusses some of the recent advances related to input features, modality and post-filtering methods for motion synthesis system, including head motion synthesis and gesture and body synthesis. Generally, there are two approaches of motion synthesis: one is rule based, and the other is data driven. A simple rule-based approach would be efficient if there is a rule, which is universal to all speakers, and no adaption is required. Rather than using rule-based approach, the data-driven approach is more suitable in this motion synthesis task because little intervention would be involved to show the speaker's individuality. In the rule-based approach's pipeline, whenever there is a demand of synthesis for a new speaker, a new set of recordings and a new set of rules would be required to boost this approach. On the other hand, the only modification in the data-driven approach is to retrain the model with the new set of data captured, which incurs a far lesser labour cost. Moreover, the model in the data-driven approach can be easily trained as speaker-independent by combining different recordings from different speakers. Additionally, we could still bootstrap a model, which is trained with less data, from a larger dataset to predict reasonable motions. This technique is called speaker adaption in speech synthesis.

In the data-driven approach, researchers chose to start with Hidden Markov Models (HMM) based for head motion synthesis (Hofer and Shimodaira [2007], Hofer et al. [2007], Ben Youssef et al. [2014], Sargin et al. [2008]) as the application of HMM is popular in speech tasks. However, the accuracy of the head motion remains low with HMMs (Hofer and Shimodaira [2007]). This can be attributed to the properties of the

HMMs and the task itself. HMMs use the Markov assumption and hidden states and determines that the next observation and next hidden state will only depend on the current state. Whereas, the motion task is many-to-many mapping and non-deterministic. This research can be considered a generation task, and neural network has been chosen to show distinguished performance in speech-based tasks (Graves et al. [2006], Graves and Jaitly [2014], Battenberg et al. [2017], Amodei et al. [2016], Zhang et al. [2017], Chan et al. [2016]). As such, this motion task is then naturally switched to neural network (Ding et al. [2015a], Haag and Shimodaira [2016], Kucherenko et al. [2019], Henter et al. [2020]). Table 2.1 summaries the differences between the neural-network-based motion research.

2.4.1 Speech-Driven Head Motion System

One of the earliest attempts to generate head motion from speech by Cassell et al. [1994] and Pelachaud et al. [1996]. Pelachaud et al. [1996] were rule based. The rules were based on the findings of Hadar et al. [1983]. For each utterance, rules are applied according to the type of utterance, specified phonemic items, stress, etc. The head motion generation was only a part in a system that attempted to synthesise a whole range of behaviour based on the text markup. They separated facial movement into phonemic, intonational, informational and affectual determinants. Head motion was mainly used as an intonational and regulating factor in the interaction. It is interesting to note that by applying the psychological findings, results of the estimated head motion can be achieved but the developed rules seem very complicated. Furthermore the interaction among the rules could lead to emergent unforeseen emergent behaviour. Finally, as with most rule-based systems, it will be hard to extend the approach or add other elements to it as it is not clear how new rules would interact with the current ones. Along similar lines, DeCarlo et al. [2002] have created a rule-based head motion synthesis module for their talking head RUTH.

Table 2.1: Summary of the recent neural-network-based head motion paper. OE:objective evaluation, SE:subjective evaluation.

Paper	Type of Motion	Type of Speech	Feature	Model Architecture	OE	SE
Ding et al. [2015a]	HeadMotion	Broadcast News	Acoustics	FNN	CCA/ACC/MSE	No
Haag and Shimodaira [2016]	HeadMotion	Conversationals	Bottleneck	Bi-LSTM	Local CCA	MOS
Sadoughi and Busso [2018b]	HeadMotion	Reading Script	F0+Energy	GAN	Parzen kernel density	MOS
Greenwood et al. [2017]	HeadMotion	Reading Script	FBanks	Variational Autoencoder	MSE	No
Kucherenko et al. [2019]	BodyMotion	Interview	MFCC	Encoder-Decoder	APE/Motion Statistics	MOS
Ginosar et al. [2019]	BodyMotion	TV shows	Waveform	GAN	L1/Correct KeyPoints	A/B Test
Yoon et al. [2019]	BodyMotion	TV shows	Text Transcript	Encoder-Decoder	No	MOS
Ghosh et al. [2017]	BodyMotion	No	Motion	LSTM+Encoder-Decoder	MSE/Classification	No
Henter et al. [2020]	BodyMotion	No	Motion	Glow	RMSE	MOS

One of the first HMM-based systems was proposed by Busso et al. [2005] and was updated in Busso et al. [2007a]. In the more recent approach, they trained HMMs on clusters of head motion built using Linde - Buzo - Gray vector quantization (Linde et al. [1980]). These are meant to represent typical head motion poses. They then picked the most likely head motion sequence based on the acoustic features. This gave the target poses that they then interpolated. Then, they added noise to create an interesting trajectory. Both their subjective and objective results have proved promising. Furthermore, using different training data, they were able to simulate different emotions. Sargin et al. [2007] developed a system that generates head motion from prosodic features. First, HMM-based clustering is performed on head motion represented as Euler angels and prosodic features separately. The correlations between the prosody and head clusters are analysed using multi stream HMMs to determine an audio-visual mapping model. The mapping model is used to generate head motion trajectories from input speech. Then, the pattern sequence is determined from the prosodic features. The associated Euler angles with the pattern sequence, then smoothed by a filter and used to drive a talking head. However, due to the deterministic property of the HMM system, the neural network system in our thesis is more suitable to apply for the task.

In addition to the modality problem, the correlation between speech feature and motion affects the prediction result as well. In order to tackle the problem of a weak link between speech and head motion, other features and their combinations have been explored. Ben Youssef et al. [2014] built HMM-based acoustic-to-articulatory inverse mapping to predict the articulatory features from speech. The estimated articulatory feature vectors were represented by the trajectories of the (x,y)-coordinates of the six active EMA coils and were then used to predict head motion through multi-stream HMMs. Their finding demonstrated that the estimated articulatory feature vectors were more correlated with head motion than acoustic features in local CCA. They also showed that the correlation between the estimated head motion using articulatory

features and original head motion (or speech) was higher than the estimated head motion using acoustic features. Ding et al. [2015a] examined LPC, MFCC and filter bank (FBank) features with deep neural network and demonstrated that the FBank-based system outperformed the MFCC-based one with a higher correlation between the predicted and original head motion. Haag and Shimodaira [2016] combined MFCC and EMA features to build bottleneck features, which were then fed to DNN-BiLSTM to predict head motion. The authors argued that contextual information is not required in training a BLSTM network because it already takes the preceding and following context into account, where combining stacked bottleneck features and a BLSTM architecture works best for objective results. Sadoughi and Busso [2018b] built a conditional GAN with BLSTM using F0, intensity (plus first and second derivatives) as the input feature to predict head motion. They claimed that the proposed system outperformed the normal BLSTM architecture models. Greenwood et al. [2017] proposed CVAE-BLSTM and used the decoder as a generative model to predict head motion, where the FBank features were used as the condition. The authors claimed that with the help of CVAE, the work solved the one-to-many problem, predicting a number of plausible motion trajectories by choosing new values for a latent space, but with the same audio features. Additionally, they reported that there is no reliable empirical measurement, and this makes the findings of this study problematic.

However, since all of the acoustic features are derived from speech waveforms, it is vital to consider the original waveforms as the inputs to fully evaluate all the information from the original observations. While using waveforms for acoustic modelling, with neural networks, is a well-researched area in automatic speech recognition (Sainath et al. [2015], Ghahremani et al. [2016], Tüske et al. [2018], von Platen et al. [2019], Loweimi et al. [2020]), to the best of our knowledge, no one has investigated the use of waveforms for speech-driven head-motion synthesis, in which a set of two data streams, speech and head motion, is dealt with rather than a single stream of speech.

2.4.2 Gesture and Body System

Kucherenko et al. [2019] implemented a speech-to-motion mapping with encoder-decoder DNN. Ginosar et al. [2019] generated motion sequence in a GAN-RNN system. The proposed generative model learnt to predict the temporal stack of poses from the given audio input, while an adversarial discriminator ensured that the predicted motion was both temporally coherent and in the style of the speaker. Yoon et al. [2019] found that the natural language was useful to predict a frame-by-frame poses with a GRU-auto-encoder. Ghosh et al. [2017] proposed a system that generates body motion recursively with a deep LSTM-RNN and de-noising auto-encoders from a given pose. Henter et al. [2020] proposed probabilistic, generative, controllable models based on normalising flows. The models were built with autoregressive and LSTMs to enable arbitrarily long time-dependencies. Objective and subjective results demonstrated that randomly-sampled motion from the proposed method outperformed task-agnostic baselines and attained a motion quality close to recorded motion capture.

2.4.3 Post-Filtering

Human motion is continuous and regular, but most of the systems predict motion in short segments due to the learning capability of deep neural networks, and there are many challenges in predicting motion from raw data over short-time horizons and long-time horizons. This shortcoming results in the predicted movements being discontinuous and either laggy or jerky. Hence, it is of paramount importance to have a de-noising/smoothing filter for these movements. There are two popular types of filters used for motion synthesis research: a linear filter (e.g., Gaussian filter, moving averaging smoothing) and a de-noising auto-encoder using a neural network (Ghosh et al. [2017]). The linear filter is a filter whose impulse response (or response to any finite length input) is of finite duration because it settles to zero in finite time. The de-noising auto-encoder is trained by inputting noisy data and computing a loss on the

output by comparing it to the ground truth data. The disadvantage of the linear filter is that it does not have the additional information/knowledge of the characteristic of the actual movement track. The linear filter uses only delayed versions of the input signal to filter the input to the output, and this may result in filtering the pivotal motion over the period, whereas the de-noising auto-encoder is trained with specific human movement data, which creates uniqueness and provides the model with knowledge of the characteristic of the movement. Thus, the keyframes of the movement have remained, while the noise is removed. It is common to apply post-processing to obtain a smooth output. Ding et al. [2015b] applied MLPG(Tokuda et al. [2000]) to generate smooth trajectories. Sadoughi and Busso [2018a] smoothed the rotations by converting rotations into quaternions and then selecting 15 key points per second, interpolating the intermediate frames (Busso et al. [2007a]), and Haag and Shimodaira [2016] applied three-order polynomial smoothing filter on the output.

2.5 Datasets

With any neural-network-based approach, a large amount of data is required. Usually, data is taken from some mocap datasets, or manipulated by methods. Needless to say, head motion systems, as they stand, use dataset-based neural network techniques to build their mapping from speech to head motion. The model presented in this thesis is no different. This section describes the data that was used for training and evaluation. First, we discuss the available datasets and then the dataset that was used.

Table 2.2: Comparison of data available in existing candidate datasets. Each column represents whether the dataset provides the specific data or in which level.

Name	Body	Clean audio	Transcription timing
IDIAP	Yes	No	None
CMU	Yes	No	None
HID	Yes	No	None
mngu0	No	Yes	Phone
ESPF	No	Yes	Sentence
IEMOCAP	Yes	No	Sentence
MPI	Yes	No	Sentence
VOCA	Yes	Yes	No
Takech	Yes	Yes	No
Trinity	Yes	Yes	Sentence
UoEMocap	Yes	Yes	Sentence

Table 2.2 summaries some existing audio/motion datasets, where each column of the table represents whether the dataset provides the type of data. These datasets have some pros and cons, which cause the suitability to our tasks in this thesis. SEMAINE (McKeown et al. [2012]) and CID (Ferré et al. [2007]) do not have motion capture, while IDIAP (Ba and Odobez [2005]), CMU (CMU) and HID (Rett and Dias [2007]) have motion capture but not audio. VOCA (Cudeiro et al. [2019]), MPI (Volkova et al. [2014]) and Takech (Takeuchi et al. [2017]) contain both audio and motion capture, and they seemed to meet our requirement. However, MPI and VOCA are too short in the total duration. Even though Takech has five hours in total, there are 1049 video clips, meaning that each clip is roughly 10 seconds long. IEMOCAP dataset (Busso and Narayanan [2007]) is the noted exception to the short dataset generalisation and for which there are 10 actors. However, it was not fully suitable for our needs as recorded an actor reading the scripts of short sentences. However, this would not affect the effectiveness if we take IEMOCAP as a benchmark. Whereas the University of Edinburgh speaker personality and mocap dataset (Haag and Shimodaira [2015]) and the Trinity dataset (Ferstl and McDonnell [2018]) contain both audio and motion capture data, and each video clip is sufficient in time. Thus, they fulfill our task requirement.

IEMOCAP contains approximately 12 hours of audiovisual data, including video, speech, motion capture of face and text transcriptions. It consists of dyadic sessions where actors perform improvisations or scripted scenarios, specifically selected to elicit emotional expressions. We only used the speech and the head motion data. We selected one of the speaker as same as Sadoughi and Busso [2018a] for benchmarking.

UoEMoCap consists of 130 non-acted natural conversational recordingsfrom 13 UK native speakers (seven females, six males). Each recording lasted approximately five minutes. The speakers were asked to act as extroverts and introverts and have non-acted natural conversations. Speech and motion data were recorded in two separate recording studios during conversations. This dataset was selected to be applied in Chapter 3 and Chapter 4.

Trinity is recorded in a motion caption studio at Trinity Colleague Dublin (Ferstl and McDonnell [2018]). It contains around four hours of recordings with a single male actor speaking freely on various topics. The actor speaks in a colloquial manner with a happy disposition and includes a large quantity of gesture motions. On top of the motion capture data, speech is also available. This dataset was selected to be applied in Chapter 5.

2.6 Methodology for Head Motion Synthesis Evaluation

Tools that can assess the quality or naturalness of predicted head motion could be very useful for research and development purposes. Such tools could speed up the development cycle and save costs from doing many listening tests. In fact, it can be difficult to compare data across head motion systems due to inconsistencies in how listening tests are conducted over the years. While authors report system performance at the time of publication, there is no guarantee that a listening test protocol is consistent from one year to the next or from one laboratory to another. Therefore, a reliable and

standardised tool could help with making comparisons and allow researchers to better characterize gains in system performance. However, developing an automatic assessment tool is beyond the scope of this thesis. Nonetheless, We strongly consider this is an important aspect to consider in the field. Moving on, the following sections discuss the measures applied in this thesis.

2.6.1 Subjective and Objective Evaluation Background

Researchers commonly use two methods to evaluate head motion systems: objective and subjective evaluation. Objective evaluation measures show how similar head motion is to the original. This evaluation indicates how well/worse the current synthetic system compares to either the existing systems or previous methods. We always assume that the gold standard is the data from motion capture.

However, objective measures only show the numerical differences between the ground truth and the generated head motion, whereas subjective testing reflects the opinions of the human observers on whether the generated motion is a close match to human-likeness. Subjective evaluation is arguably effective as humans forms their own judgements of those motion. However, subjective testing is more time-consuming and costly. Therefore, researchers usually use the objective evaluation in the first stage and decide whether or not it is worth proceeding with subjective testing for their systems.

In the following section, we mainly discuss the factors of the subjective evaluation in motion synthesis and explain the reason why we chose such measures in our experiments. The objective measures are discussed in the experiment chapter.

2.6.2 Subjective Evaluation

2.6.2.1 Types of Subjective Evaluation and Their Prior Use

In this section, we describe the subjective evaluation method that researchers usually perform on their head motion system. Head motion synthesis is animated with some 3D software by inputting standard motion file or by joining frames of pictures together to form a video. Since our research presents 3D models, we will only discuss and present the corresponding methods for this type of animation. For animating appropriate synthesis, there are commonly three types of methods that have been applied in the literature. The most common rendering method is to visualise a virtual avatar that has human-like skin, eye colour, mouth and hair. Extensive research has proven the effectiveness of this method (Choi et al. [2001], Busso et al. [2005], Sargin et al. [2008], Le et al. [2012]). A less common method is to visualise a virtual avatar but no human-like elements, i.e., smooth shaded. (Campbell [2000], Munhall et al. [2004], Hofer [2010], and Kucherenko et al. [2021]) are examples of this method. Last, silhouettes were used for head motion analysis in Hadar et al. [1983] study, and this study is widely used in head motion research. Yet silhouettes have not been used in subjective evaluation. Thus, it is worth to try whether silhouettes are useful for this purpose.

The A/B test and Mean Opinion Score (MOS) test are two common evaluation methods used in speech-related synthesis tasks such as TTS and head motion synthesis.

In an A/B test, participants are usually asked to select a preference from two samples, and they can also refrain from choosing either of the two samples. However, in an A/B forced test, participants are obliged to select one of the samples.

On the other hand, the MOS test requires the participants to give a score for each sample individually on a fixed scale between 0 to 5 or 10 points. The scale of the score represents the quality of the sample from bad at zero to excellent at the other extreme.

One variation of a MOS test is called the MUltiple Stimuli with Hidden Reference and

Anchor (MUSHRA) test, and it is recommended by the International Telecommunication Union (ITU) for evaluating similar quality signals. The MUSHRA test requires the participants to rate each sample by moving a continuous slider. On the slider, there are regions marked as Bad, Poor, Fair, Good and Excellent. All of the samples from different systems are shown together. The participants are allowed to listen/watch the samples repeatedly until they have given the scores for all the samples. This is different from a MOS test as a sample is given at each time to be scored. Moreover, depending on the implementation of the evaluation, the participants may be asked to give scores that can distinguish the differences between samples.

A labelled ground truth is provided for distinguishing the differences among the samples in a MUSHRA test. Furthermore, besides the samples generated from systems, a 'hidden reference' and a 'hidden anchor' are included in the sample set, meaning that within the sample set, the reference signal is provided again. The anchor refers to the worst case sample among the sample set. These two hidden samples should be the upper and lower bounds in the rated scores of a sample set. In head motion synthesis, we can use the original motion capture as the reference. Whereas a low pass filtered version of the reference should be created as an anchor, and this is recommended by the ITU.

The original ITU recommendations are based on testing quality of audio modification, and an example application would be testing compression algorithms. However, there is no specification for using MUSHRA on speech-related tasks. Thus, it is not necessary to follow all the recommendations in our head motion synthesis.

The problem with MUSHRA and MOS testing is that they are prone to bias, which can originate from many different sources (Zieliński et al. [2007]). These problems can be resolved by following the recommended practice when we design the interface and experimental conditions. Additionally, a researcher should obtain a sufficient number

of samples from all of the subjects for statistical confidence.

The main objective in the subjective evaluation for our task was to find out the largest difference in the scores to distinguish the good and the bad, not the highest possible score. It is not useful if the participant gave similar high scores to an animation with no motion and the motion capture data.

2.6.2.2 Considerations for Subjective Evaluation

As we stated above, challenges may arise in the subjective evaluation. It could be impossible to eliminate all of them. Thus, to ensure that the task is bias-free and effective, we limited ourselves to some of the factors that we thought were important. Most importantly, they could be mitigated by experimental design. These factors are as follows:

- What type of animation should be used?
- How long does the training phase need to be?
- How long can the test last before the participants become bored and stop paying attention?
- Can only native speakers of the language be used as participants?
- Does the listening environment significantly impact the results?

In the following, these factors are discussed in more detail. The famous uncanny valley effect (Seyama and Nagayama [2007]) could mean that exceedingly realistic animation might be considered 'creepy'. While with a less realistic animation, the participants can focus on the movement. However, if the animation is too unrealistic, the participants may not be able to tell the difference between good and bad animation.

Hofer [2010] demonstrated that the type of animation significantly impacts the subjective evaluation by comparing three main types of animation in the subjective eval-

uation: 1) animation using motion capture, 2) synthesised motion and 3)randomly generated head motion. However, he did not compare his results to head motion that looks natural but is not synchronised to speech. In fact, this randomly generated head motion looked very unnatural, and participants could easily point that out, and this makes the lower bound of the result to be uncertain.

Campbell [2000] and Park et al. [2002] showed that eye and lip movement easily catch people's attention and significantly impact the perception of naturalness. Thus, these two movements have to be excluded from the animation or else the participants take their quality into account for the final results. If the eye and lip motion are the same for all types of head motion, the participants will always take the movement of eye and lip into account when scoring the head motion. This could affect the final evaluation result. Hofer [2010] included the lip and eye motion in the head motion synthesis, and his results could be influenced by them.

Kucherenko et al. [2021] organised a gesture synthesis workshop called GENEA (Kucherenko et al. [2020]), and they used the results from each team and designed a formal subjective study online. In the study, they used a modified version of MUSHRA to evaluate the generated motion in terms of human-likeness and naturalness. Further, Jonell et al. [2020] compared the effectiveness of taking the subjective study in-lab or online. Their results demonstrate no difference between the two participant pools regarding their evaluations of the gesture generation models and their reliability scores.

In fact, head motion is expressed unconsciously; thus, when it comes to judgement, people find it difficult to decide what is right or wrong. Thus, back to this type of subjective evaluation, people are not capable of distinguishing whether a head movement is appropriate to the speech. Additionally, participants also have to be familiar with the evaluation interface. In speech-related synthesis research, this is always achieved by providing a training phase before the real evaluation starts Benoît et al. [1996]. How-

ever, head motion synthesis is new and differs from speech synthesis tasks, it could be a problem for us to decide how long the training phase should be. This training phase is also included in other speech-related tasks and is recommended in ITU MUSHRA. These points reinforce our choice to include the training phase in our head motion evaluation.

Mental fatigue during difficult tasks is a well-studied field. Van Orden et al. [2000] stated that the quality of the performance would get worse in visual tasks of long periods. Since head motion evaluation is considered a type of visual task as well, this means that the participants find it increasingly difficult to distinguish the quality of the head motion over time. Persson et al. [2007] found that fatigue impacts 'interference tasks'. This refers to the process of filtering out relevant data during the comparison between systems, thus further affecting the final evaluation results. This effect is not only psychological; in this regard, using an EEG (Boksem et al. [2005]) were able to show that participants' attention decreased when fatigue increased. This decreasing attention was impossible to prevent using any method with correct instructions. Nevertheless, all of the participants would probably face this problem. The above studies suggest that it is not good to design the experiment to be too long. Even though there is no direct study reporting fatigue during head motion evaluation, it is important to determine the most optimal length of the evaluation.

It would be much more convenient to conduct head motion synthesis evaluation over the internet than bringing participants into a laboratory. However, Reips [2002] warned that changes in the environment are among the factors causing the results to be skewed. Kittur et al. [2008] also stated that some experiments are not suitable to be conducted online. However, Buchholz and Latorre [2011] had a different opinion, pointing out that preference testing could be suitably conducted online. Moreover, Wolters et al. [2010] also found that speech quality testing could be conducted online, even though different headphones and speakers could affect the quality of the audio tracks. Last,

Jonell et al. [2020] demonstrated that results did not differ between in-lab and online testing for gesture synthesis evaluation. Based on the studies above, we believe that changes in the environment impact the results, although the impact may not be significant. If the distributions of both results collected from different environments are similar, we may perform some reasonable transformation to the results, and this could eliminate the effect of the impact of the environment changes. Thus, the different conditions can be controlled for.

Finally, we discuss the culture. Graham and Argyle [1975] studied the cultural impact on which gesture people produce while speaking. Kita [2009] showed that native language always has a larger influence on the types of gestures people produce. However, regardless of the different gestures produced, there is no direct link between natural gestures and culture. In effect, we always considered culture as a factor when we designed the evaluation. Since our dataset consists only of native English speakers, it would be convenient to hire native English speakers to perform the evaluation experiment.

2.6.3 Analysis and Discussion

The above sections presented our findings about subjective evaluation methods. In terms of subjective evaluation, by utilising a modified version of MUSHRA testing, we found that the current practice of using realistic avatars was the best approach for evaluating head motion. This is because during comparative testing, participants are reliably able to tell when head motion is synchronised.

2.7 Chapter Summary

In this chapter, we have presented the key background needed for understanding this thesis. In particular, we covered different methods of representing rotation and ex-

plained reason for our choice of using rotation vectors. We also presented some background information on different types of speech features, including EMA features, which are estimated from lower level speech features. Moreover we discussed some SOTA techniques in deep learning for speech-driven head motion that are used in the presented system, namely CorrNet. Finally, we described and explained why we chose subjective evaluation for motion synthesis.

Chapter 3

Speech-Driven Head Motion System with Waveform

3.1 Introduction

Estimating head motion from speech has been investigated with different acoustic features or combinations of them as mentioned in Chapter 2. This aims to provide information on different aspects for better estimation results. For example, MFCC does not have prosodic information and thus a combination with F0 to achieve better results in other speech-related tasks (Zhou et al. [2010], Hasija et al. [2022]). However, since the waveform is the stem of all these acoustic features, it should contain full information of the speech. No previous study has investigated estimating head motion from waveform directly. Moreover, due to the information loss in the handcrafted process of the acoustic features, the correlation between the features and head motion remains low.

Unlike speech-driven head motion tasks, using waveform directly has been proposed in ASR tasks. Sainath et al. [2015] stated that a waveform-based acoustic model matches the performance of log-mel filterbank energies when used with a SOTA CLDNN acoustic model. Ghahremani et al. [2016] found that adding a feature extractor in a neural

network can achieve SOTA results with waveform. Tüske et al. [2018] suggested that adding a second level of time-convolutional element on top of the Gammatone feature extraction pipeline (Schluter et al. [2007]) could boost the performance of the acoustic model further with waveform. von Platen et al. [2019] ascertained that a multi-span CNN with different span of raw waveform signal outperforms a FBank-based acoustic model and showed that this multi-span CNNs learnt filters that were rather different from the log mel-filters. These methods prove the effectiveness of applying waveform in ASR tasks. Thus, we hypothesise that a similar performance boost occurs when we use waveform in head motion estimation, since both ASR tasks and head motion estimation are similar as both use speech as input.

We have to consider a few questions when applying waveform in head motion estimation. First, what form of waveform should we apply in the task? Tüske et al. [2014] have already proved that direct waveform input to the neural network would make the learning more difficult for ASR compared to the hand-crafted features. Thus, head motion estimation would be more difficult because ASR task is dealing with only one data stream, whereas head motion estimation handles two data streams. The above ASR studies have already provided a solution to this: feature extractor or, in other words, bottleneck feature. Second, what information of waveform should we use in the task? As it is different from ASR, head motion estimation does not require full information of speech. Head motion has been shown to be related to pitch accents (Graf et al. [2002]) and strongly linked to speech features especially F0 (Kuratate et al. [1999]). However, the linguistic information is not required for head motion because Mcneill [1987] demonstrated that the 'gesture lead' phenomenon arises. Therefore, we have to distill the information within speech. Third, how are the estimated motion processed? Before discussing what method to post-process, we must explain why the output should be de-noised. One of the many possible reasons is that the frequency of speech and the frequency of head motion are different. A complete head

motion is measured to be last at least 400ms (Hofer and Shimodaira [2007]), whereas most people speak at an average speed of four to five syllables per second. This frequency difference requires rapid predictions within a short-time horizon and causes the predicted motions to be discontinuous and laggy/jerky when combining together. Another possible reason is the limited learning capability of the neural network, which encounters challenges in learning data over a long-period (e.g., vanishing gradient). Forth, how can the proposed system be evaluated? Standard measures such as MSE only show the differences between the recorded ground truth and the predicted motion. However, there are many possible combinations of movement patterns (e.g., different angle of movement) that occur with the same speech content regardless if the person is speaking or listening. Thus, such measures have a limitation of misleading researchers by resulting in a significant difference between the recorded ground truth and the predicted motion; however, the predicted motion shows the same movements with a larger degree only. So far, there is no such measure to evaluate movement along with the time axis. The reason is that it is difficult to imagine whether a movement occurs at a specific timing based on the above measures. Thus, it is hard to evaluate a system based on the existing objective measures.

To overcome the problems of high dimensionality and irrelevant information, we propose a canonical-correlation-constrained autoencoder (CCCAE) to extract low-dimension features from raw waveforms, where hidden layers are trained not only to minimise the error of encoding and decoding but also to maximise the canonical correlation with head motion. The extracted features of a low dimension are then fed to another neural network for regression to predict head motion. Then for the post-filtering issue, we resolve it by training a de-noising auto-encoder with clean data instead of dropout/noisy data. The reason for not using linear filters is that the linear filters are based on identifying the impulse transfer function that satisfies the requirements of the filter specification, whereas our proposed filter requires inputting clean data to train and learn in

reconstructing smooth head motion. Last, we propose a new objective measure called term-weighted Value (TWV). TWV is used to measure the quality of a detection system, and it is useful in our regression system as the rotation vector is formulated by angle and rotation matrix. Assuming that a typical movement creates a large change in the angle, TWV can be used to measure the angle peaks for the generated head motions. Therefore, TWV is useful to visualise whether a movement occurs at a right timing. We showed that the features obtained with the proposed approach are more useful for head motion prediction than those with a standard autoencoder. We evaluated the new approach through comparisons with other acoustic features objectively.

3.2 Related Work

In head motion synthesis, representation learning has been proposed to build the bottleneck feature from acoustic features.(Haag and Shimodaira [2016]; Fares et al. [2022]). Although these bottleneck features yield improved results, they learnt to contain the relationship between the acoustic and head motion data. However, the model does not consider the full information from speech itself. As discussed previously, the handcrafted features suffers from the issue of the information loss. It is natural to consider the original waveforms as the input to neural networks so that we can fully make use of the information in the original observations.

Furthermore, the objective of these bottleneck features usually takes the construction differences between the ground truth and the estimation only. This applies to the variety of the head motions from the same speech, leading to low correlation between features and targets. DCCAE was proposed by (Chandar et al. [2016]; Wang et al. [2015]) to effectively model two data streams with construction error and CCA loss, and the models were applied to cross-language tasks and multi-view feature learning, where reasonably high correlations between two data streams are expected.

Then, come to the output, the predicted head motion is always noisy or jerky as discussed. Thus, a de-noising filter is always a part of the motion synthesis system in the literature. To the best of our knowledge, the common way to create noisy data is either applying dropout to the 'clean' data for making the data discontinuous or add Gaussian noise to the 'clean' data. Ghosh et al. [2017] proposes a de-noising auto-encoder using a neural network by dropping out the joints of the body skeleton. Due to there is only one joint (x, y, z) in our head motion task, dropping out any one of the trajectories does not simulate the natural noisy head movements. On the other hand, adding Gaussian noise to create noisy data does not yield expected jerky movements as they would naturally occur. Thus, training an auto-encoder with Gaussian noise data would not be effective. The difference in our proposed approach is to learn the natural head motion instead.

In the evaluation process, Haag and Shimodaira [2016] proposed using local CCA to calculate the correlation between the predicted motion and the ground truth. Kucherenko et al. [2019] then proposed calculating the velocity, acceleration and jerkness of the movement for both the predicted motion and ground truth, instead of showing the numerical differences only. Both studies performed subjective evaluation to show the effectiveness of the their proposed systems.

3.3 Methodology

Our proposed system can be separated into three modules; (1) a canonical-correlation-constrained autoencoder (CCCAE) for compressing the high-dimensional waveform input to distributed embedding of low dimensions; (2) a regression model for predicting the head motion from the compressed embedding; (3) a post-filtering autoencoder for reconstructing smooth head motion. The overall framework of our proposed model is shown in Figure 3.1.

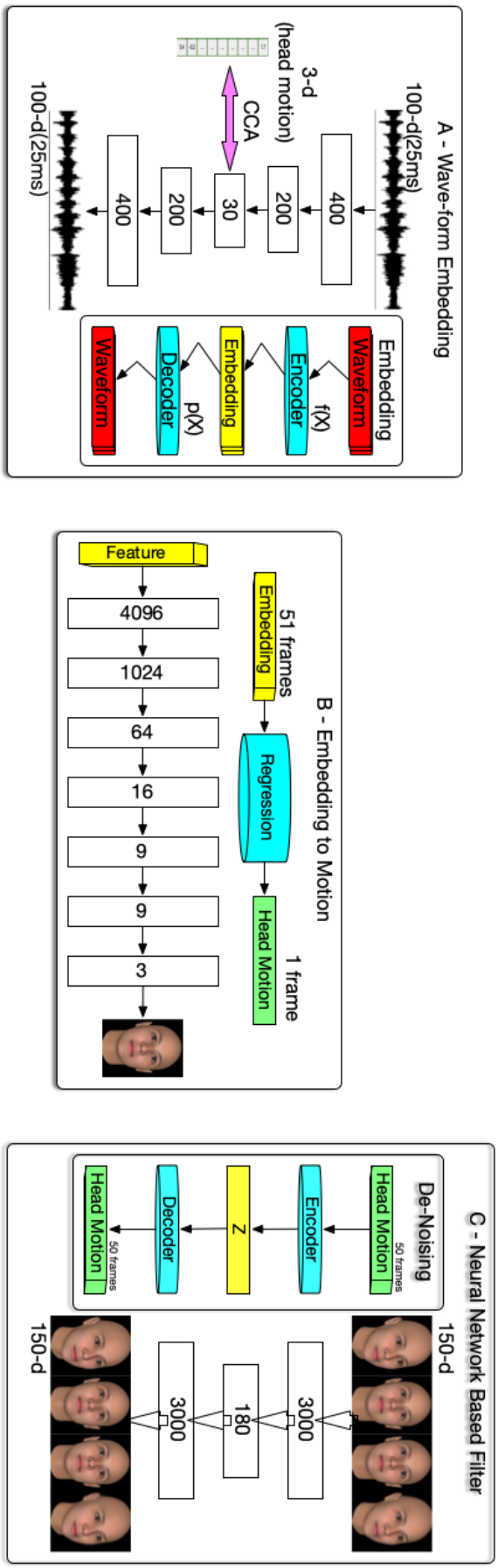


Figure 3.1: Overview of the proposed system comprised of three modules: (A) waveform embedding with CCAE, (B) DNN-based head motion regression from the embedded features, (C) post-filter with an autoencoder.

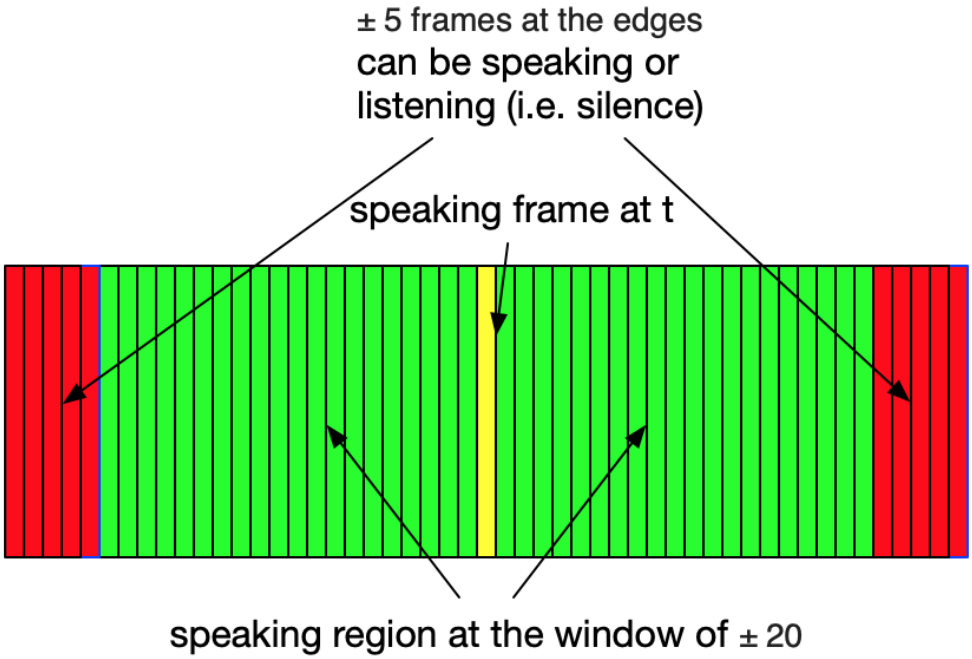


Figure 3.2: The 51 frames representation of the speech feature. Yellow represents the speaking frame at $time_t$, green represents the speaking regions at $time_{t\pm20}$, red represents the ±5 edges and can be either speaking or listening (i.e., silence)

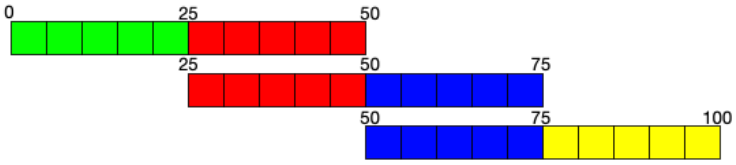


Figure 3.3: Number of distinct head motions in one second. Same colors represent the overlapping when shifting. The value in the figure represents the number of frames of head motion, and each frame is 10ms

3.3.1 Waveform Embedding

As discussed in Chapter 2, standard AE with reconstruction error only does not get any explicit learning signal encouraging it to share the capacity of its common hidden layer between the views, since the views are not guaranteed to be projected to a common subspace. When we apply the resulting embedded features to the downstream task, the final score is not guaranteed to be good since the information compressed in the embedding is not controllable. Then, researchers started to explore combining the training procedures of the AE and downstream tasks and achieved surpassing results.

However, we argue that there is a trade-off between the compression and task scores, which makes the training more difficult. Thus, we propose to use CorrNet in our work, which combines the advantages of standard AE-based and CCA-based approaches. This framework of autoencoder for a set of two data streams was proposed by Chandar et al. [2016] and Wang et al. [2015]. The framework by Wang et al. [2015] consists of two autoencoders and optimises the combination of canonical correlation between the learned 'bottleneck' representations and the reconstruction errors of the autoencoders. Since head motion is parameterised with a time series of rotation vectors of three dimensions in the present study, we did not need to use an autoencoder to reduce the dimensionality further. We thus employed a single autoencoder, in which hidden layers were trained in such a way as to not only minimise the reconstruction error but also maximise the canonical correlation with head motion. Thus, instead of projecting the two features to a common subspace, we projected raw waveforms to a subspace so that the embedded features are well correlated with head motion. In our assumption, we make use of the data in speaking regions to train the proposed CCCAE because human produces active motion during speaking. In such cases, the trained model supposes to only encode correlated information with the head motion from the original waveform to the embedded feature.

We trained the proposed CCCAE with the following objective function:

$$\text{Obj}_{\text{CCCAE}} = \sum_t \|\boldsymbol{X}_t - p(f(\boldsymbol{X}_t))\|^2 - \alpha \text{CCA}\,(f(\boldsymbol{X}), \boldsymbol{Y}) \tag{3.1}$$

where $\boldsymbol{X}_t \in \mathbb{R}^{100\times 1}$ represents the input raw waveform vector at a time instance t to the encoder, $f(\,)$ represents the projection with the encoder, $p(\,)$ represents the reconstruction with the decoder, $\boldsymbol{X} \in \mathbb{R}^{100\times T}$ and $\boldsymbol{Y} \in \mathbb{R}^{3\times T}$ denote the whole sequences of waveform vectors and head motion vectors, respectively. In our case, $\mathbf{Y}$ is represented as 3D rotation vector, discussed in Chapter 2. Each of the dimensions can

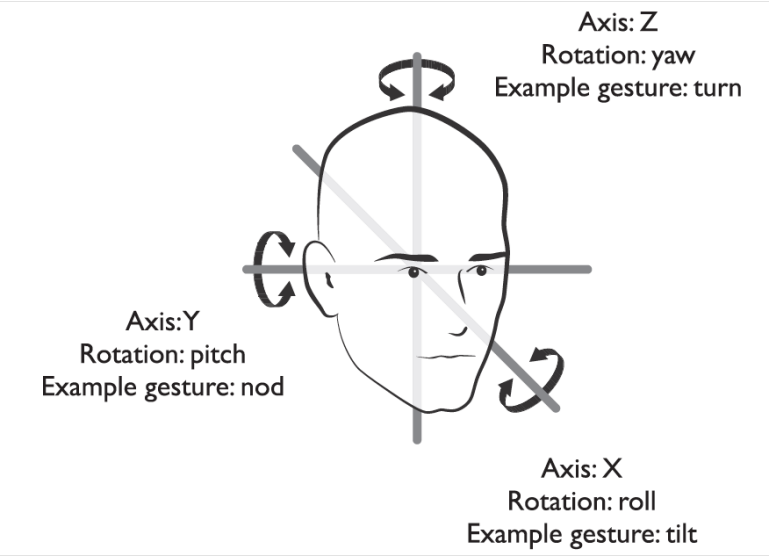


Figure 3.4: Schematic overview of rotations along three axes, as well as example movements most frequently used in communicative head gesturing (Wagner et al. [2014])

swing individually to act as a movement (shown in Figure 3.4). CCA( ) is the canonical correlation function. $\alpha \geq 0$ is the weighting factor, where $\alpha = 0$ corresponds to a standard autoencoder with an MSE loss function. In our later experiment, we set $\alpha = 0$ for standard AE and $\alpha = 1$ for our proposed CCCAE.

3.3.2 Head Motion Regression

A simple feed-forward deep neural network was applied here for the regression from the waveform embedded features to head motion. We do not consider more complex models such as CNN and LSTM, because the present study focuses on a compact and efficient representation of speech features rather than the regression of head motion, and previous studies (Ding et al. [2015a], Haag and Shimodaira [2016]) showed no significant differences among the models. Accordingly, we also did not consider auto-regressive models such as WaveNet (van den Oord et al. [2016]).

As shown in Figure 3.2, a context window of ± 25 frames, which is equivalent to 525 ms effective speech content, was employed to predict head motion parameters.

3.3.3 Post-filter

Since the output trajectories of our neural networks are noisy or discontinuous due to the nature of speech, we applied a post-filter as post-processing to obtain smooth head motion trajectories for animation. We built a neural-network-based de-noising autoencoder following the architecture, trained with the 'clean' data (Lu and Shimodaira [2019]). The common training procedure of the de-noising model, which comprises applying dropout/Gaussian noise to the clean data for recreating noisy data Ghosh et al. [2017] Vincent et al. [2010], does not work with our model as the Gaussian noise method does not give the expected jerky movements as they would naturally occur. The dropout method, on the other hand, drops one of the three trajectories of the head motion, and this strictly limits the movement, causing unnatural behaviour. Therefore, instead of removing the noise from the jerky head motion, we expect the de-noising filter to learn and know how the smooth head motion over a period should be. We assume a complete head motion in every consecutive 500ms Hofer and Shimodaira [2007] time frame, as the input, M_{in}, to the de-noising filter and the output, M_{out}, are of the same length. We follow the architecture in Ghosh et al. [2017], using the feed-forward neural network, trained with the back-propagation learning algorithm, but as the input dimension is different in these two cases, we explored the best depth and width of the model for recovering the head motion. Overall, the filter can be represented by the following architecture:

$$M_{out} = W_{dl}(W_{el}M_{in} + b_{el}) + b_{dl} \quad \text{for} \quad 1 \leq l < L \tag{3.2}$$

where e represents the encoder operator, d sets the decoder operator, and l is the number of layers.

3.4 Experimental Setup

For the purpose of our experiments, we selected data in UoEMocap from three males (Subject A, Subject B, Subject C) and three females (Subject D, Subject E, Subject F). Six recordings (around 30 minutes) of each subject were used for training, two (around 10 minutes) for validation, and the remaining two (around 10 minutes) for evaluation, ignoring the differences in terms of the speaking style. We trained our models for each subject. Note that speaker-dependent training is a common practice in speech-driven head motion synthesis (Busso et al. [2005], Ding et al. [2015a], Sadoughi and Busso [2018b]).

Speech Features. Audio in the database was recorded with a headset microphone at 44.1 kHz with 32-bit depth and a MOTU 8 pre mixer (MOT). Separate recording channels were used for the two speakers, and a synchronisation signal was recorded on a third channel in the mixer. For the purpose of this work, the audio signal was down-sampled to 4 kHz prior to feature extraction. It is because the matrix calculation for the CCA objective increases dramatically if the size of the matrix increases, and we only have limited computational power. Raw waveform vectors were extracted using 25 ms windows with 10 ms shifting, which resulted in 100 dimensions. A total of 13 MFCCs were formed by combining one energy coefficient and twelve mel-cepstral coefficients, using SPTK (Yoshimura et al. [2023]). We also added their first and second-order derivatives, resulting in 39 MFCCs. Voicing probability and energy were computed using openSMILE (Eyben et al. [2010]) and smoothed with a moving average filter with a window length of 10 frames. All the features were normalised in terms of variance for each dimension.

Head Motion Features. Movements of the head as a 3D rigid body were recorded with the NaturalPoint Optitrack (Nat) motion capture system at a 100 Hz sampling rate. From the marker coordinates, rotation matrices for the head motion were computed

using singular value decomposition (Soderkvist and Wedin [1994]), which were further converted to rotation vectors of three dimensions. Furthermore, we assumed that there is a complete head motion in every consecutive 500 ms and 250 ms shifting to ensure smoothness and continuity in every distinct head motion as shown in Figure 3.3.

In training, we only used the frames where the target speaker for head motion prediction was speaking so that the models learnt the relationship between speech and head motion properly. In the evaluation, we made use of all the input audio sequences to generate head motion parameters.

The following notations were used in the rest experiments:

- Wav_{AE}: Embedded features extracted from the standard autoencoder (i.e., the output of proposed CCCAE with $\alpha = 0$)
- $\text{Wav}_{\text{CCCAE}}$: Embedded features extracted from the proposed CCCAE with $\alpha = 1$
- M_{MFCC}: Regression model trained with MFCC feature
- M_{AE}: Regression model trained with Wav_{AE}
- M_{CCCAE}: Regression model trained with $\text{Wav}_{\text{CCCAE}}$

M_{MFCC}, M_{AE} and M_{CCCAE} use the same architecture in Figure 5.5(B) to predict head motion, while each model takes different feature vectors as input.

Training was conducted on a GPU machine and a multi-CPU machine with Tensorflow version 1.12 by mini-batch training using Adam optimisation (learning rate 0.0002) (Kingma and Ba [2015]). We also employed layer-wide pre-training (Takaki and Yamagishi [2016]).

In the evaluation, test data of the same speaker was fed into the trained regression model, and head motion was predicted frame by frame. After that, the output of the prediction model was then combined to form distinct head motion of 50 frames, which

were fed to the post-filtering autoencoder. The final output for animation was generated with the overlap-add method. Moreover, IEMOCAP dataset was used for benchmarking.

3.5 Objective Measures

Objective measures can demonstrate the individual performance of different layers of the hierarchy and show if the approach is sound before subjective testing. The followings discuss available objective measures and their suitability for this thesis. Then, we also propose our own objective measure that better reflects the performance of a motion synthesis system than the current practice.

3.5.1 Canonical Correlation Analysis and Head Motion Synthesis

One of the most commonly used correlation tests for two streams of multivariate data, such as head motion trajectories and speech features, is CCA (Alpert and Peterson [1972]; Lambert and Durand [1975], Haag and Shimodaira [2016], Lu and Shimodaira [2020]). This measure was introduced by Hotelling [1936] as an extension of Pearson's correlation, which calculates the correlation between scalar, for calculating the correlation in multi-dimension vectors. The idea is to map two streams of data, which may not be of the same width, onto a common hyperplane and then find the Pearson's correlation between vectors in that plane. For two streams of multivariate data arranged into a matrix, where each of the rows corresponds to one observation, $\mathbf{X} \in \mathbb{R}^{n \times T}$ and $\mathbf{Y} \in \mathbb{R}^{m \times T}$, and cor is the Pearson's correlation function, the canonical correlation score $\rho^{(c)}$ is defined as follows:

$$\rho^{(c)} = \mathrm{cor}(\mathbf{a}'\mathbf{X}, \mathbf{b}'\mathbf{Y}) \tag{3.3}$$

where **a** is the $n*1$ vector and **b** is the $m*1$ vector that satisfy the following:

$$\mathbf{a},\mathbf{b} = \arg\max_{a,b} \mathrm{cor}(\mathbf{a}'\mathbf{X},\mathbf{b}'\mathbf{Y}) \tag{3.4}$$

It is also useful to define the following:

$$U_1 = \mathbf{a}'\mathbf{X} \tag{3.5}$$

$$V_1 = \mathbf{b}'\mathbf{Y} \tag{3.6}$$

This process is then repeated with the added constraint that new **a** and **b** are uncorrelated with the first and so on. The complete set of these vectors is **A** and **B**. The complete sets of **U** and **V** are known as canonical variables or scores. These vectors lie on the hyperplane mentioned above. **X** is mapped onto the hyperplane by **A**, and **Y** is mapped onto the hyperplane by **B**.

We employed local canonical correlation analysis (local CCA) as suggested by Haag and Shimodaira [2016]. The difference between global CCA and local CCA is that global CCA measures the correlation over the whole sequence, whereas local CCA only calculates the sub-sequence's CCA score within a time window and then takes the mean value of all the obtained scores. The reason for selecting local CCA is that there is rarely linear correlation held over long sequences, which is instead calculated by global CCA, as the head motion trajectories change over time. We used a time window of 300 frames or 3 seconds with a 50% overlap. We used the following formula to calculate local CCA:

$$\begin{gathered} \mathrm{T} = \{0, 150, 300, ..., T_n\} \\ \mathrm{r}_{Average} = \frac{1}{|T|}\frac{1}{d}\left(\sum_{t\in T}\sum_{i=1}^{d} \mathrm{corr}\left(A^{[i]}X_{[t:t+n-1]},\ B^{[i]}Y_{[t:t+n-1]}\right)\right) \end{gathered} \tag{3.7}$$

where $A^{[i]}$, $B^{[i]}$ are the i's canonical coefficients obtained in the global CCA, d is the

dimension of features and T is the length of the utterances.

3.5.2 Motion Peak Detection

Term-weighted value (TWV) is used to measure the quality of a detection system, and it is useful in our regression system as the rotation vector is formulated by angle and rotation matrix. Assuming that a typical movement creates a significant change in angle, TWV can be used in measuring the angle peaks for the generated head motions. We used the following formula to calculate TWV that specifies the trade-off between misses and false alarms (Fiscus et al. [1970]):

$$\mathrm{TWV}(\delta) = 1 - average\mathrm{P}_{miss} + \beta * \mathrm{P}_{FA}(term, \delta) \tag{3.8}$$

$$\beta = \frac{C}{V} * (\mathrm{Pr}_{term}^{-1} - 1) \tag{3.9}$$

where the $\frac{C}{V}$ is the cost/value ratio and this is set as 0.1. Pr_{term} is the prior probability of a term. In the original paper, Pr_{term} is set at 10^{-4}, but in our case, we set it at 10^{-2}. That is due the following reasons: 1) Our speech utterances are much shorter than in the original paper. 2) There are not many true peaks in the ground truth, thus putting high penalty in the false alarm for the system would deleverage the effect of missing term as the predicted head motion seems always more active than the ground truth. 3) Our peak matching windows is 51 frames, whereas the test utterance file has about 30+k frames. The total peak matching frames (51 * the average peaks in Subjects) only covers less than 10% of the speech utterance. Whereas in the spoken term detection task of the original paper, there are terms in every frame; thus, using 10% of the original penalty looks reasonable.

The false alarm generally is defined as alarm systems in many different applications being triggered by something other than the expected trigger event. In our case, the unexpected trigger event refers to there being no motion movement in the reference/ground

truth, but motion movements occur in the predicted. The miss values then refer to the opposite way, in which there is no motion movement in the predicted one, but occur in the reference/ground truth. Then to calculate the miss and false alarm, we applied the following:

$$\mathrm{P}_{miss}(term,\delta) = 1 - \frac{\mathrm{N}_{correct(term,\delta)}}{\mathrm{N}_{true}(term)} \tag{3.10}$$

$$\mathrm{P}_{FA}(term,\delta) = \frac{\mathrm{N}_{spurious}(term,\delta)}{\mathrm{N}_{NT}(term)} \tag{3.11}$$

$$\mathrm{N}_{NT}(term) = \mathrm{n}_{tps} * \mathrm{T}_{speech} - \mathrm{N}_{true}(term) \tag{3.12}$$

$$\tag{3.13}$$

where:

- $term$ is the condition we would like to be detected, in our case, it is the angle peak
- δ is the detection threshold, which refers to the time window
- $\mathrm{N}_{true}(term)$ is the total number of the angle peak in the ground truth
- $\mathrm{N}_{NT}(term)$ is the number of opportunities for incorrect detection of $term$ in the corpus (='Non-Target' $term$ trails)
- $\mathrm{N}_{correct(term,\delta)}$ is the number of correct (true) detections of $term$ with a detection score greater than or equal to δ
- $\mathrm{N}_{spurious}(term,\delta)$ is the number of spurious (incorrect) detections of $term$ with a detection score greater than or equal to δ
- n_{tps} is the number of trials per second of speech (arbitrarily set to 1)
- T_{speech} is the total amount of speech in the test data (in frames)

The maximum possible TWV is 1.0, corresponding to 'perfect', and the value of 0.0

means nothing. Negative TWVs are possible when large numbers of false alarms happen. Moreover, the cost of a false alarm is effectively constant across all the $term$, $\approx 1/T_{speech}$, since in practice $T_{speech} >> N_{true}(term)$, while the cost of a miss is variable and depends on the number of true occurrences of all the $term$, $= 1/N_{true}(term)$(Weg-mann et al. [2013]).

3.5.3 Velocity, Acceleration, Jerk

Calculating the absolute differences in positions between the generated motion and the motion capture does not justify for natural motion (Kucherenko et al. [2019]). Having similar distribution is another factor that we should consider for a plausible candidate for natural motion. Plausible motions do not require measures such as speed or jerk to closely follow the original motion, but they should produce a similar distribution. That is why we would like to study distribution statistics, namely velocity, acceleration, and jerk. They are calculated by taking a finite difference between joint positions at time t and $t-1$ and the derivative of joint positions.

3.5.4 KL divergence

KL divergence is used to measure the similarity between two probability distributions. It is useful in our evaluation because it shows whether there is capacity for common patterns in the acoustic features and personal behavior of the different subjects. Such personal behaviors would result in distinct pattern distribution in the later motion generation. We use the following formula to calculate the KL divergence:

$$\mathrm{D}_{KL}(P||Q) = \sum_{x \in X} P(x) \log \left(\frac{P(x)}{Q(x)} \right) \tag{3.14}$$

where P, Q are the two probability distributions, which are defined on the same probability space, X. In the later evaluation, we applied symmetric KL divergence, which is

defined below:

$$\text{Symmetric}_{KL} = \text{D}_{KL}(P||Q) + \text{D}_{KL}(Q||P) \tag{3.15}$$

3.6 Results and Discussion

We first train the proposed DCCCAE with different embedding sizes to determine the structure of the model. Then the defined embedding is used to analyse and compare with those existing acoustic features. We then select few of the features based on the analysis results to train the regression model. Lastly, the regression results are presented with different objective measures mentioned in Section 3.5.

3.6.1 Autoencoder Reconstruction

Table 3.1: Comparison of different widths of $\text{Wav}_{\text{CCCAE}}$, where NMSE and local CCA are calculated between $\text{Wav}_{\text{CCCAE}}$ and the original head motion for Subject A in UoEMocap.

Width	NMSE			CCA		
	Train	Valid	Test	Train	Valid	Test
15	0.411	0.507	0.480	0.245	0.216	0.219
30	0.173	0.239	0.221	0.264	0.234	0.248
60	0.233	0.261	0.250	0.220	0.194	0.194

High dimensionality has been affecting the popularity of the usage of waveform as the input to neural networks, even though the waveform contains the original information of the acoustic features. Here, we seek to resolve this problem by using our proposed model, CCCAE. In the previous section, CCCAE has been described as not only reducing the dimension of the input feature effectively but also maximising the correlation between the embedding feature and the target. In this experiment, we built the DCCCAE model as we described in Fig 3.1(a). We first explored the possible dimensions of embedding features with sizes of 15, 30 and 60 in the middle layer and the rest of the layers were unchanged, where the original dimension of waveforms input was 100.

This could give us a clear idea about the trade-off between the recovery of waveforms and the correlating information.

Looking at Table 3.1, which shows the comparison of different widths of the proposed embedded feature in terms of NMSE and local CCA for training, validation and test dataset. The result of the validation set demonstrates that the higher the dimension of the embedding feature is, the better the recovery of the waveform is. It is clear that the size 15 is the worst in terms of recovering the waveform as there is too little information. On the other hand, the size 60 is the least correlated to the head motion because there is still too much irrelevant information. Overall, the results show that the size of sample 30 is the best choice to provide the clearest results. The result of the test set is provided as well but is not involved in selecting the architecture. With the test set results, we can notice that the selected size 30 achieves the lowest NMSE and highest CCA. It is interesting that the size 60 is supposed to have the lowest NMSE in our hypothesis, which contains the most information and has a similar dimension to the waveform input. This could be explained that there is a trade-off between the reconstruction and CCA loss, the alpha value in Eq 3.1 should be tuned for better generalisation. However, we did not put much effort into this exploration.

3.6.2 Feature Analysis

In the introduction, we hypothesised that since waveforms contain full information of speech, there is some irrelevant information hindering the learning generalisation of the system. Thus, to better understand the relationship between the head motions and the information within waveforms or some common features (acoustic or prosody), a basic correlation analysis of local CCA was carried out between speech features and head motions before the regression training and evaluation.

Results of local CCA for each speech feature and for each subject are displayed in Figure 3.5. The findings suggest that F0+Energy gives the lowest score, and MFCC

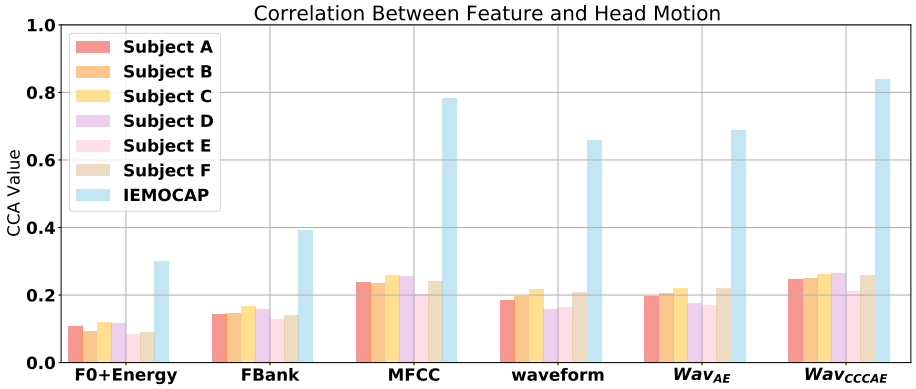


Figure 3.5: Local CCA between speech features and original head motion for the test set.

achieves the highest in the handcrafted features, and $\text{Wav}_{\text{CCCAE}}$ achieves the highest in all the features. One of the possible reasons why F0+Energy achieves the worst result is that this feature only contains prosodic information regarding the loudness of the sound. The loudness of the sound only affects the amplitude of the motion, but does not affect the change of the motion Kuratate et al. [1999]. Comparing the waveform and the proposed feature, we can see a large improvement (at least 30% is achieved on average) in the results of the test set with $\text{Wav}_{\text{CCCAE}}$ for each subject, but only a small improvement is noted with Wav_{AE}.

In the meantime, we used an external dataset, IEMOCAP (Busso and Narayanan [2007]), to evaluate our proposed model. We then calculated a correlation between MFCC and head motion that is similar to the findings reported by Busso and Narayanan [2007]. Our proposed feature has an improvement of 6% for MFCC and 27% for waveform.

As we aim to build speaker-dependent models, we would like to understand how each feature related to the subjects, examining whether there are common patterns. We visualised the features using T-SNE. Observing Figure 3.6, we noticed the pattern that the sparser the point that each subject's feature is, the lower the CCA between the feature and the head motion. Moreover, Wav_{AE} and $\text{Wav}_{\text{CCCAE}}$ show the effects of gathering

these points compared with waveforms. This gathering effect shows that the models extract latent features that capture the underlying explanatory factors for the observed input Mohamed et al. [2022], Bengio et al. [2013]. This also shows in Table 3.2, which shows the average symmetric KL divergence between the subjects. In each feature, the smaller the value is, the larger the overlapped area of the distributions. Since KL divergence is another way to calculate the similarity between the distributions, it could further highlight how the features could be related to head motion. For the calculation, we took the feature from all the combinations of every two subjects as P and Q in Equation (3.14) to calculate the symmetric KL divergence and averaged the results. In terms of the values, we can observe that the value of $\text{Wav}_{\text{CCCAE}}$ is much larger than Wav_{AE}, this refers that a smaller overlapped area for $\text{Wav}_{\text{CCCAE}}$. Thus, we believe that $\text{Wav}_{\text{CCCAE}}$ has a better effect than Wav_{AE} showing a clear distinct cluster for each subject. We also notice that FBank feature has the largest value in this KL divergence result, but it has the second lowest correlation in Figure 3.5. This implies that little speaker-independent information is carried in the feature. It is because those bank pass filters are designated to capture the information related to the human vocal tract, which is one of the main distinguishing characteristics of individual Chougule et al. [2014].

We also assume that each subject has their own person-dependent mannerisms and this affects the head movement in multiple ways, but there are still some patterns of head movement that remain unchanged in all subjects. With the CCA loss objective, $\text{Wav}_{\text{CCCAE}}$ shows a well-organised and distinct distribution of each subject's feature data as there are some feature points where subjects overlapping each other (key properties of the head movement were not changed), and some feature points are spread in different directions (this was person dependent). This distribution has not been shown in the graph of any other feature. We believe that the overlapped areas show the properties of the head motion amongst all the subjects. As the correlation between this feature and head motion is still unclear, future academic studies could develop these

areas of research. People may argue that seeing a cluster is a good thing or not. Since each subject model is trained independently, showing a cluster means that the model has learnt the personality based on the speech and head motion input. This leads the downstream task to be designated for the particular style. Otherwise, not seeing a cluster refers to the model only capable of learning general information about speech and head motion, not related to personal. This should not be what we expect from.

Table 3.2: Average symmetric KL divergence over subjects to indicate the similarity of the feature distribution in all dimensions, examining whether there is a common pattern in the acoustic feature among subjects

Measure \ Feature	F0+Energy	FBank	MFCC	Waveform	Wav_{AE}	$\text{Wav}_{\text{CCCAE}}$
Symmetric KL	6.59	11.87	6.00	5.69	6.96	8.98

Table 3.3: Comparison of different systems in terms of performance of head motion prediction, where NMSE and local CCA are calculated between predicted head motion and ground truth.

System	Subject	Training		Test	
		NMSE	CCA	NMSE	CCA
$\text{M}_{waveform}$	A	1.02	0.12	1.56	0.24
	D	2.71	0.08	2.44	0.16
M_{MFCC}	A	0.78	**0.49**	1.42	**0.41**
	D	**0.55**	**0.57**	1.55	**0.42**
M_{AE}	A	1.00	0.17	**1.06**	0.21
	D	1.15	0.09	**1.14**	0.09
M_{CCCAE}	A	**0.55**	0.42	1.39	0.35
	D	0.66	0.39	1.24	0.32

3.6.3 Head Motion Estimation Results

In addition to performing feature analysis, which was presented in the previous section, we also investigated the effectiveness of the feature by building a neural network to predict head motion using those audio features. In this section, we built a simple FNN to generate head motion and then evaluate it with different objective measures. We selected MFCC, Wav_{AE} and $\text{Wav}_{\text{CCCAE}}$, which were outstanding in the basic analysis, to use in the later evaluation of the regression models in the following section.

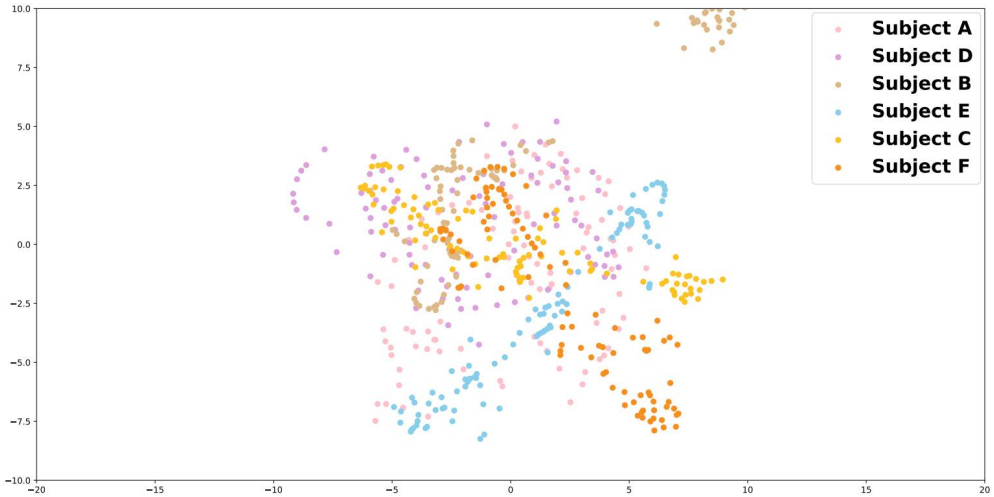
Additionally, we built a baseline model in the first experiment, which uses waveform as an input, to validate the improvement of the proposed feature.

3.6.3.1 NMSE and Local CCA

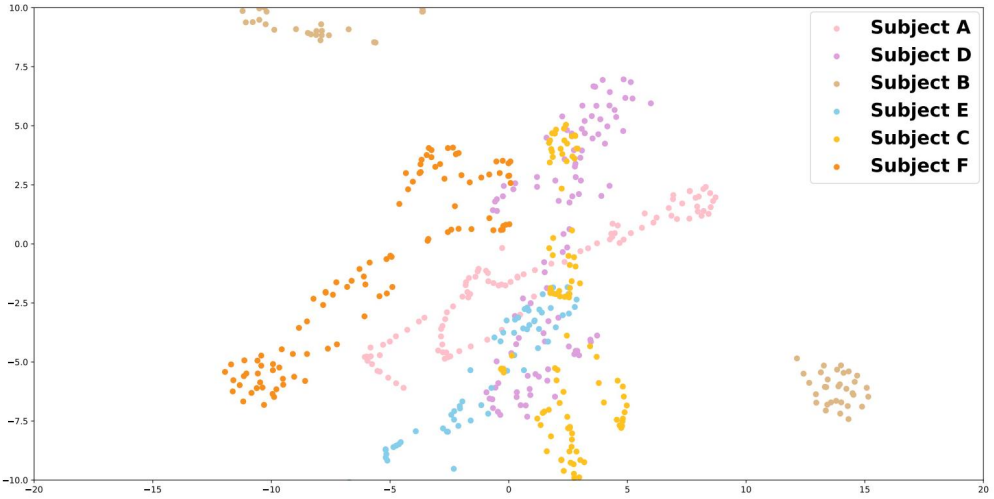
First, we compared the effectiveness of estimating head motion with different crafted features to the baseline model. Since we are using the same architecture, the difference here is just the input layer of the model. The results in Table 3.3 show the comparison of the system with different input features in terms of performance of head motion estimation. $M_{waveform}$ achieves the worst result in two subjects reflecting that those crafted features outperform this baseline model. Moreover, $M_{waveform}$ gets similar local CCA results as M_{AE}, but a larger NMSE. This refers that $M_{waveform}$ produces more head motions with speech waveform than M_{AE}, however, those head motions are not correlated to the ground truth at all. Even though the baseline model was trained with full information of speech, the results reflect that the model has difficulties in dealing with the information, so hardly to be generalised. This was also proven by (Tüske et al. [2014]), who found that direct waveform as input to the model is difficult to be trained well.

Next, since the crafted features had outstanding improvement compared to the baseline model, we increased the number of speakers to seven and continued the comparison within the crafted features. Figure 3.7 reveals how NMSE and local CCA with the ground truth (original head motion) are involved in an FNN system trained with different features, which are used to investigate the evaluation of predicted motion. Another coping strategy, which was expected to seek a chance score, was also developed on the grounds of well-computed local CCA between existing motion and randomized sequences that characterize totally different and unsynchronized subjects. The hypothesized chance score for the subjects is shown in Table 3.4.

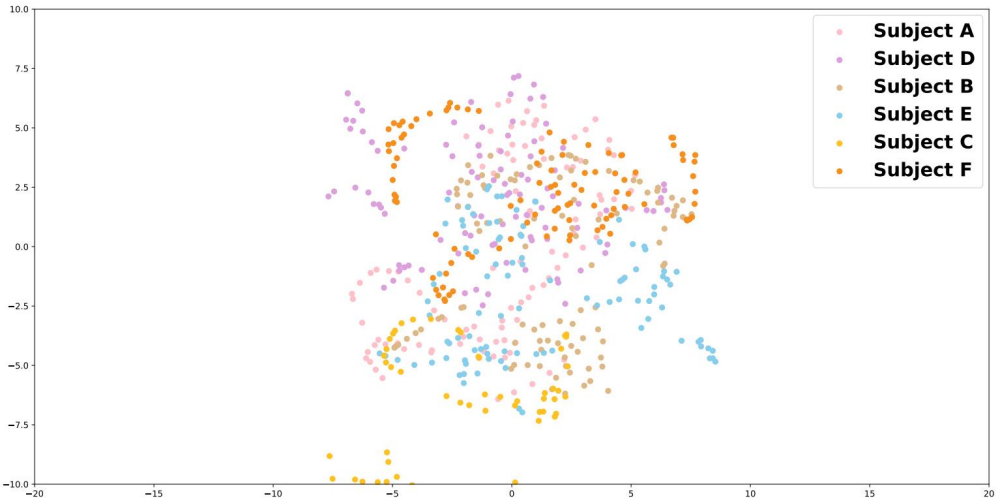
It is notable that, regardless of the lowest NMSE, the result of M_{AE} could be biased.



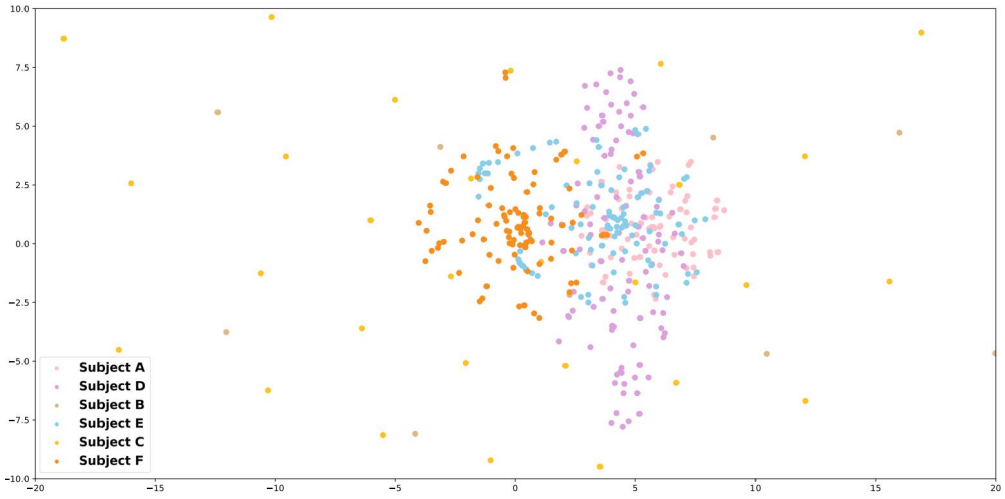
(a) F0 + Energy



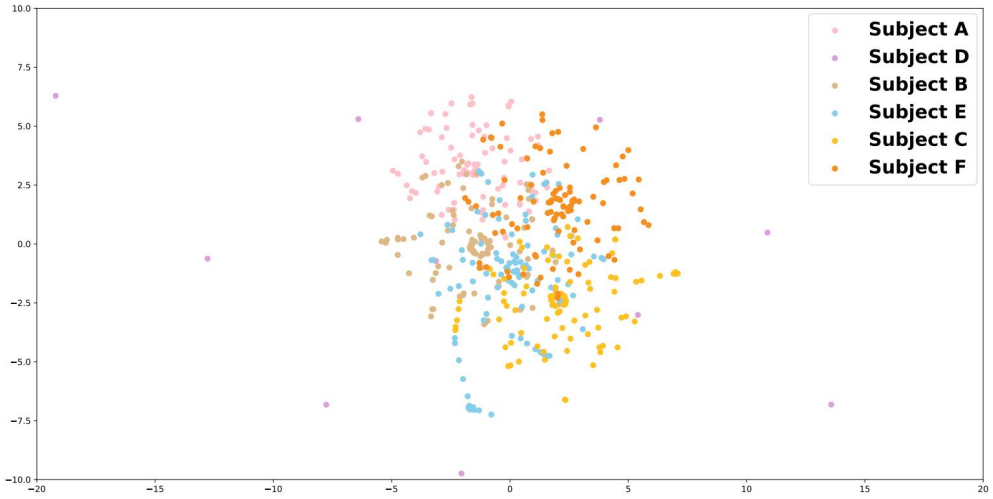
(b) FBank



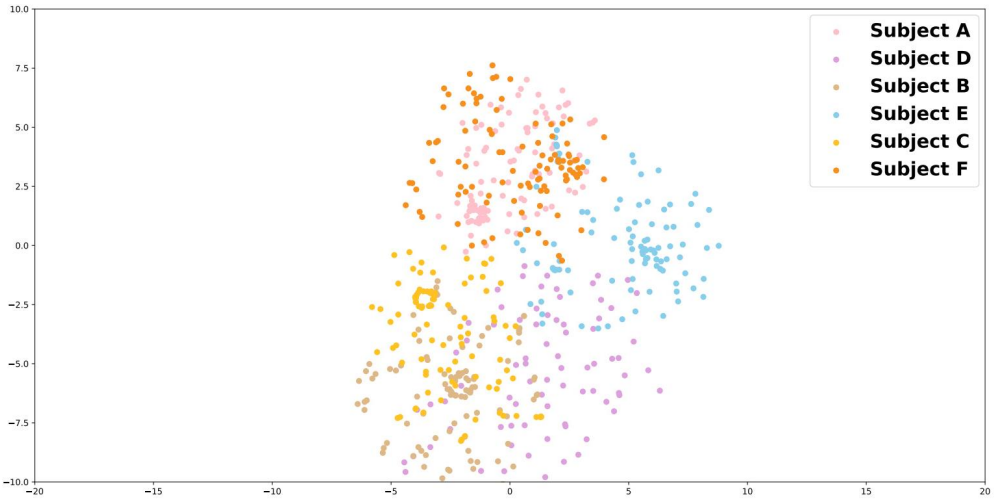
(c) MFCC



(d) Waveform

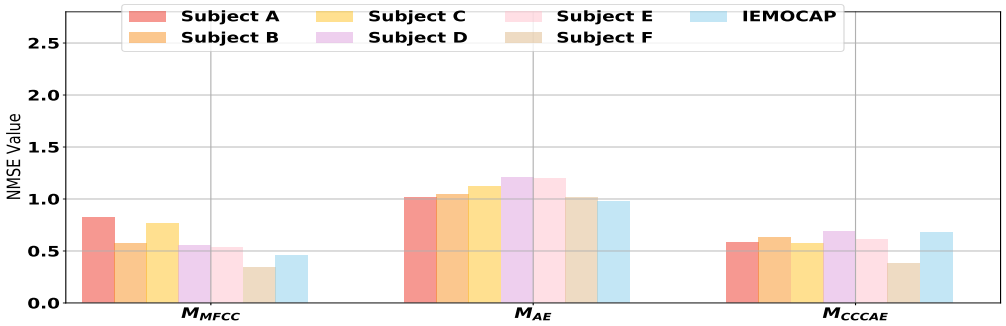


(e) Wav_{AE}

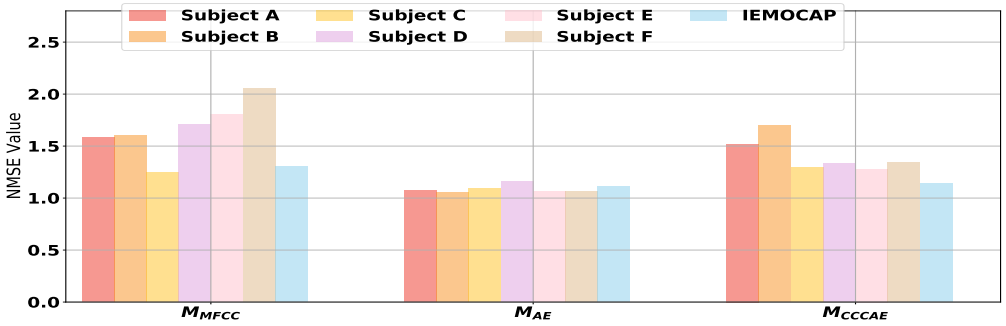


(f) $\text{Wav}_{\text{CCCAE}}$

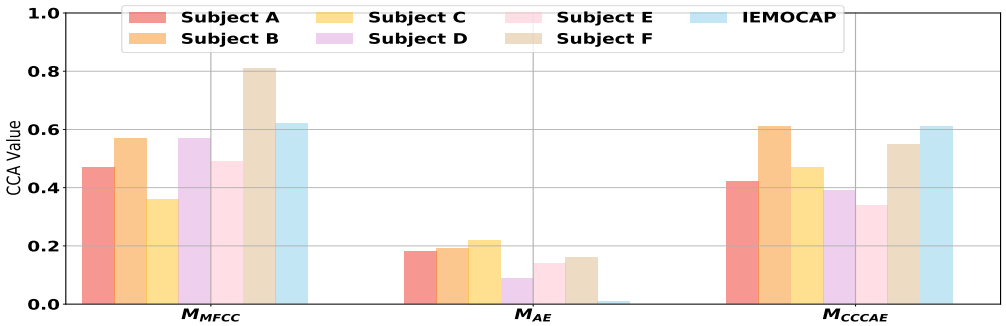
Figure 3.6: T-SNE visualisation of the feature distribution for Subjects A-F to visualise whether there is a common pattern in the head motion among subjects



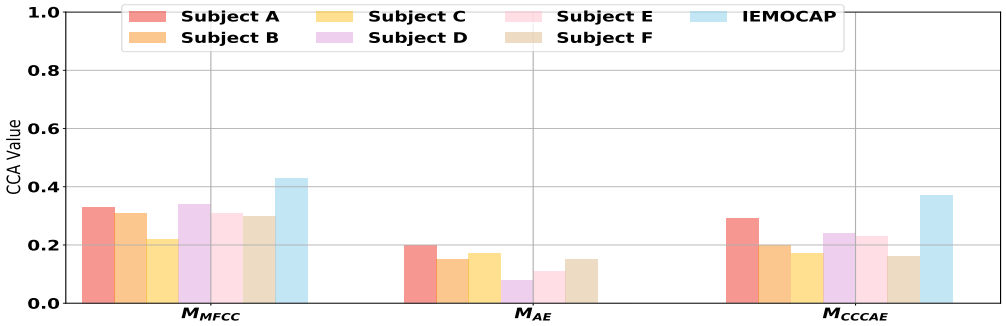
(a) NMSE Training



(b) NMSE Test

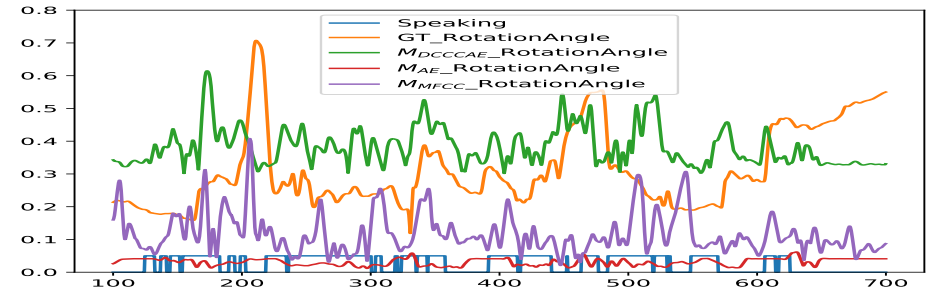


(c) CCA Training

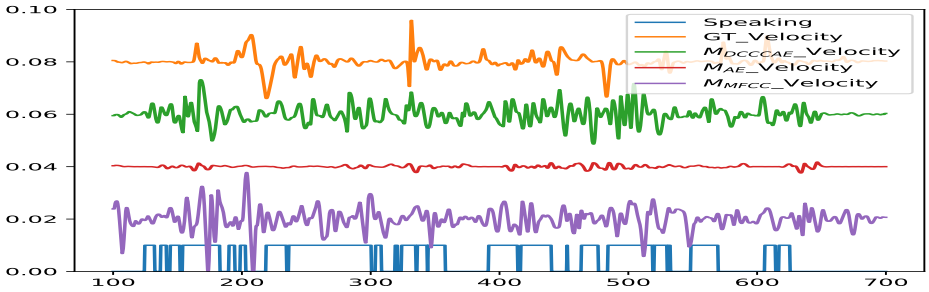


(d) CCA Test

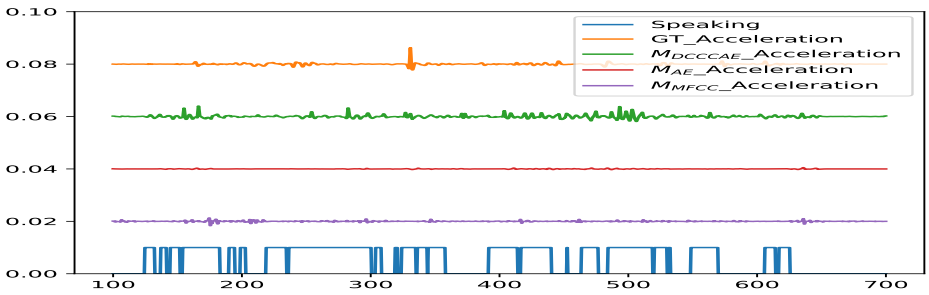
Figure 3.7: Comparison of different features in terms of performance of head motion prediction for different subjects, where NMSE (Figure a and b) and local CCA (Figure c and d) are calculated between predicted head motion and ground truth.



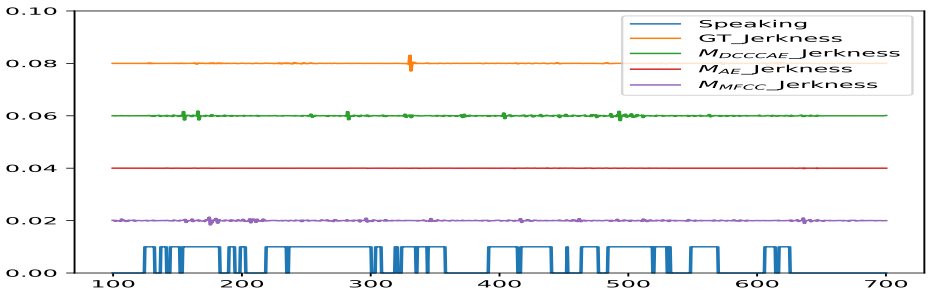
(a) Rotation Angle



(b) Velocity



(c) Acceleration



(c) Jerkness

Figure 3.8: Comparison of different features in terms of Velocity, Acceleration, Jerkness of head motion for Subject A.

Table 3.4: The local CCA between the ground truth and randomised sequences of another subject, showing the lowest bound of the CCA between two head motion streams.

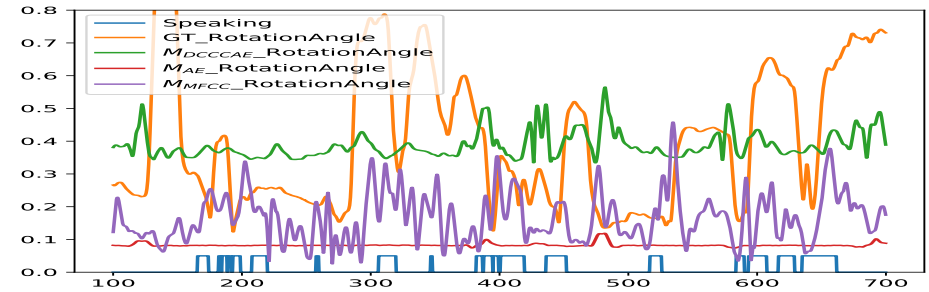
Measure \ Subject	A	B	C	D	E	F	IEMOCAP
Unsynchronised CCA	0.14	0.11	0.11	0.11	0.10	0.10	0.12

Little movement of predicted head motion directly results in NMSE being close to 1.0. This explains why the chance score mechanism is better than M_{AE} for all subjects. M_{CCCAE} has a better performance for most of the subjects except Subject B and Subject C in terms of NMSE. However, M_{MFCC} achieved the highest local CCA for all subjects. This suggests that M_{CCCAE} and M_{MFCC} have different strengths in different metric domains. Overall, the local CCA of M_{MFCC} and M_{CCCAE} in the test dataset is higher than the chance scores.

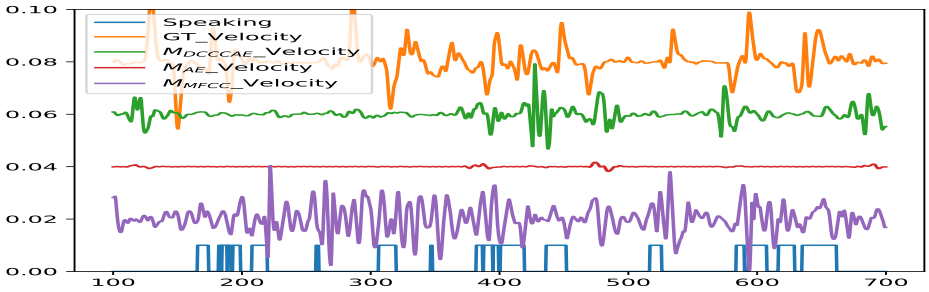
3.6.3.2 Velocity, Acceleration, Jerkness

Besides showing the numerical differences between the generated motion and the ground truth, the generated motion must have the right acceleration, whereas too fast or too slow motion does not look natural (Kucherenko et al. [2019]).

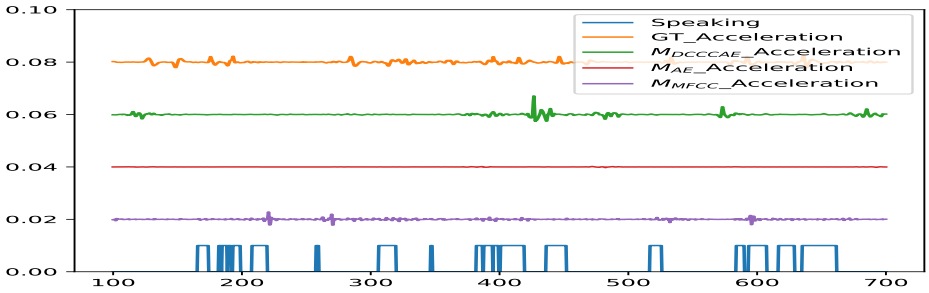
Figure 3.8 and Figure 3.9 display the changes of rotation angle, velocity, acceleration and jerkness during conversation for two speakers trained with the proposed and baseline models . Figure 3.8(a) shows the changes of rotation angle. M_{AE} has the lowest changes over time, which indicates that the animated head looks still. M_{CCCAE} demonstrates fewer changes than M_{MFCC} and is much more similar to the ground truth. Similarly, regarding the velocity in Figure 3.8, a higher frequency of change is observed in M_{MFCC} than M_{CCCAE} and the ground truth. However, regarding the acceleration in Figure 3.8, M_{CCCAE} displays a higher frequency of change than others, and the ground truth has a peak in the frequency of change. Last, Figure 3.8(d) illustrates the jerkness. M_{CCCAE} and M_{MFCC} display more changes than the ground truth. Overall, Figure 3.9 for Speaker D shows similar properties as Figure 3.8 for Speaker A.



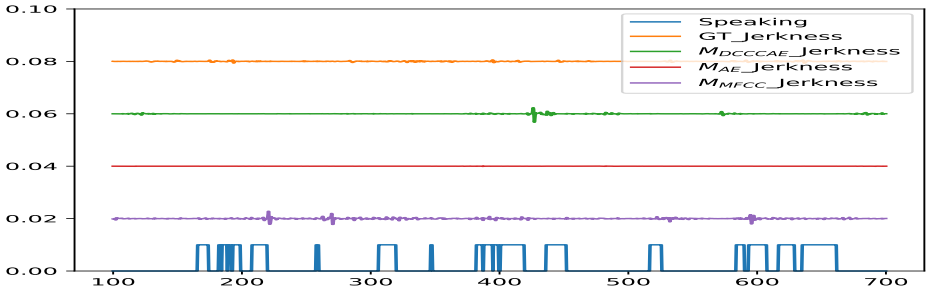
(a) Rotation Angle



(b) Velocity



(c) Acceleration



(c) Jerkness

Figure 3.9: Comparison of different features in terms of Velocity, Acceleration, Jerkness of head motion for Subject B

Table 3.5: Comparison of different systems using TWV in terms of the matching of the angle peak detection.

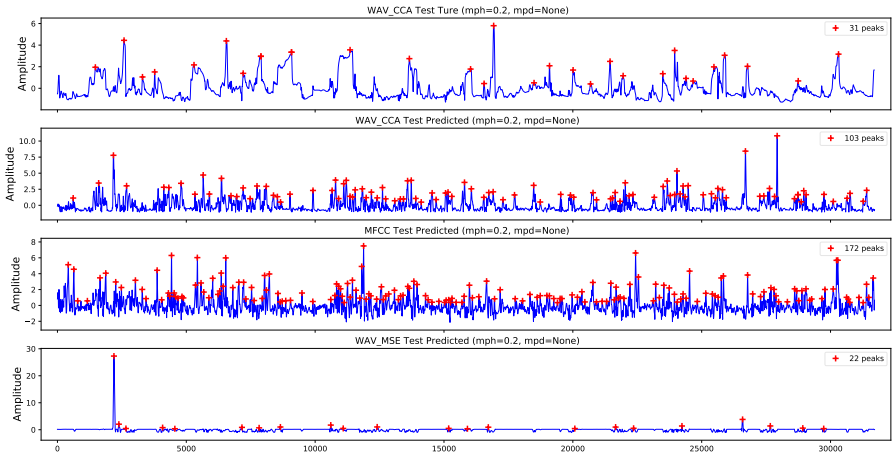
Subject	System	P_{miss}	P_{FA}	TWV
A	M_{MFCC}	0.677	0.005	0.269
	M_{AE}	1	0.001	-0.007
	M_{CCCAE}	0.677	0.003	**0.292**
B	M_{MFCC}	0.789	0.004	**0.172**
	M_{AE}	0.895	0.002	0.089
	M_{CCCAE}	0.842	0.002	0.137
C	M_{MFCC}	0.696	0.007	**0.236**
	M_{AE}	0.913	0.003	0.059
	M_{CCCAE}	0.696	0.007	0.232
D	M_{MFCC}	0.711	0.005	**0.236**
	M_{AE}	0.867	0.001	0.121
	M_{CCCAE}	0.822	0.002	0.155
E	M_{MFCC}	0.718	0.005	**0.231**
	M_{AE}	0.821	0.003	0.151
	M_{CCCAE}	0.744	0.006	0.200
F	M_{MFCC}	0.65	0.008	0.275
	M_{AE}	0.8	0.003	0.169
	M_{CCCAE}	0.55	0.008	**0.365**

3.6.3.3 Peak Detection

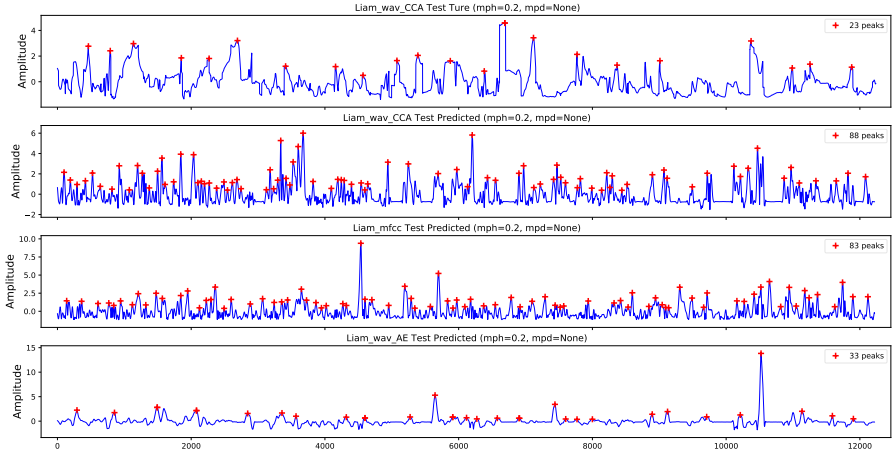
In addition to the analysis of the levels of the trajectories, we also applied an angle peak matching between the generated head motion and the ground truth. First, we calculated the motion angle from the rotation vector and then applied AMPD algorithm (Fiscus et al. [1970]) to detect the peak occurrence over time as shown in Figure 3.10 and Figure 3.11. Next, we applied a fixed sliding windows (51 frames) between the ground truth and the estimated head motion simultaneously to check for a matching peak in the window. Also, we used a window of 21 frames to prevent duplicate matching.

Table 3.5 displays the comparison of different systems, where the angle peak detection tradeoff value was computed between the generated head motion and the ground truth. Even though M_{AE} has the lowest false alarm and the highest missing values, this reflect a fact that there is little movement (or no movement) generated by M_{AE}, which is also shown in the NMSE analysis (near 1.0). Overall, M_{AE} has the lowest TWV values. Our

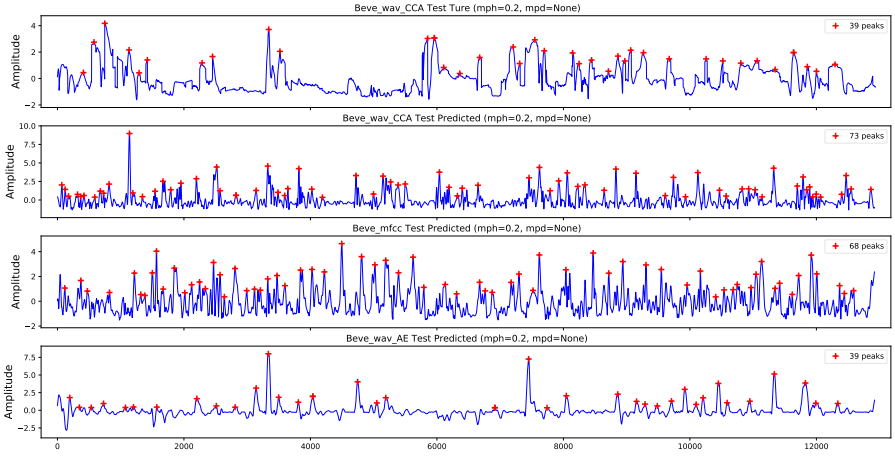
proposed model, M_{CCCAE}, achieves significant improvement over M_{AE} by the CCA constraint in training, where it achieves the highest TWV in Subject A and Subject F. The reason why of M_{CCCAE} were still out beaten in some subjects by M_{MFCC} might be that M_{CCCAE} generates still movement in the silence region, whereas M_{MFCC} outputs head motions simultaneously as the ground truth, thus increasing the possibility of matching.



(a)Subject A peak detection value, 31(GT), 103(M_{CCCAE}), 172(M_{MFCC}), 22(M_{AE})

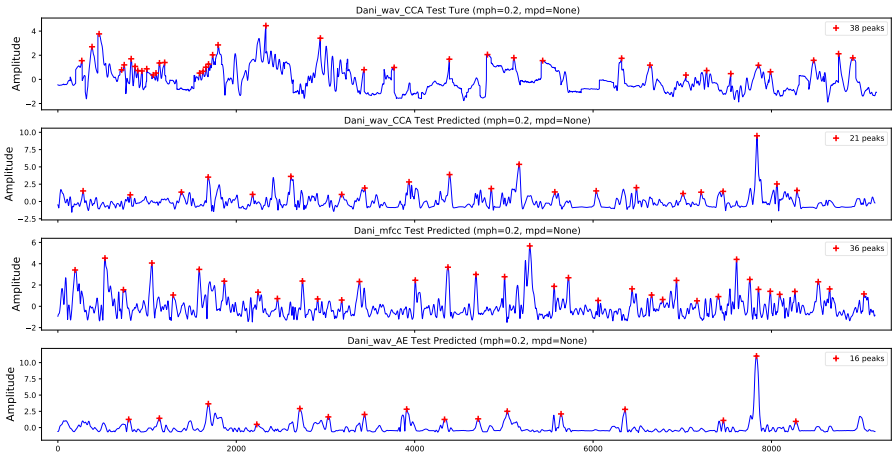


(b)Subject B peak detection value, 23(GT), 88(M_{CCCAE}), 83(M_{MFCC}), 33(M_{AE})

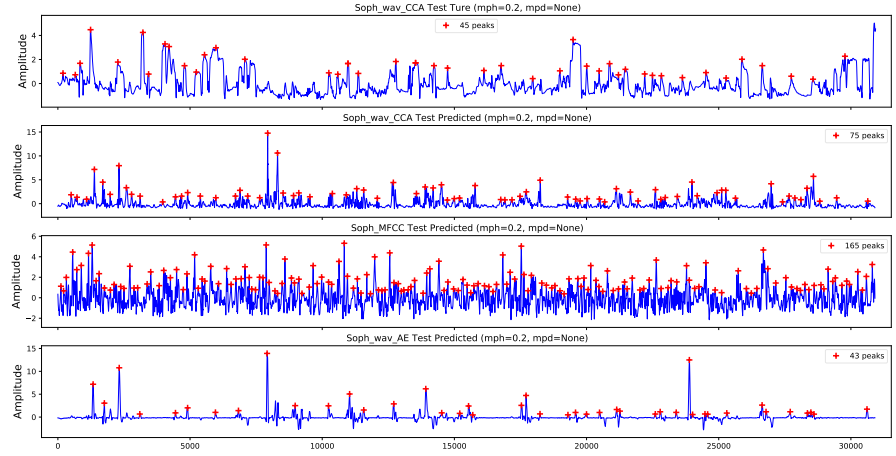


(c)Subject C peak detection value, 39(GT), 73(M_{CCCAE}), 68(M_{MFCC}), 39(M_{AE})

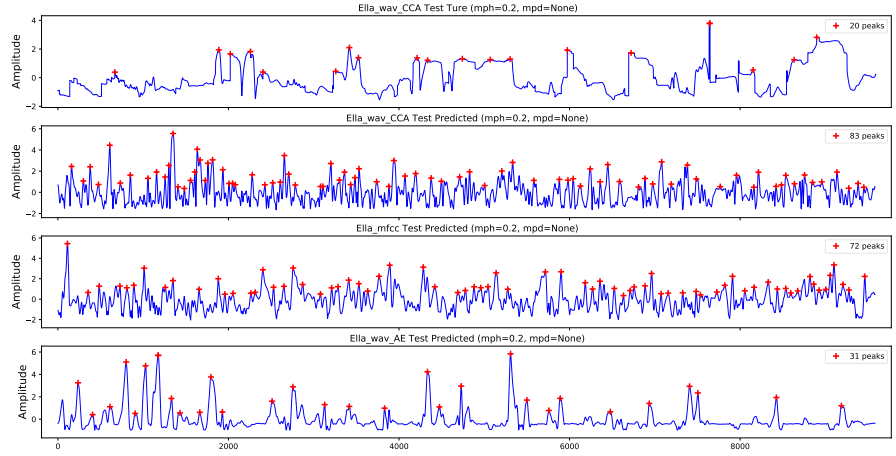
Figure 3.10: Detection of angle peak for each subject. The peak is detected by automatic multiscale-based peak detection (AMPD) Scholkmann et al. [2012].



(d)Subject D peak detection value, 38(GT), 21(M_{CCCAE}), 36(M_{MFCC}), 16(M_{AE})



(e)Subject E peak detection value, 45(GT), 75(M_{CCCAE}), 165(M_{MFCC}), 43(M_{AE})



(f)Subject F peak detection value, 20(GT), 83(M_{CCCAE}), 72(M_{MFCC}), 31(M_{AE})

Figure 3.11: Detection of angle peak for each subject. The peak is detected by automatic multiscale-based peak detection (AMPD) Scholkmann et al. [2012].

3.7 Further Considerations

While the proposed M_{CCCAE} alone did not outperform M_{MFCC} in all objective aspects, there were performance improvements compared to the baseline model and M_{AE}. Specifically, M_{CCCAE} produced more smoothness and lesser quick motion than M_{MFCC} from the motion distribution figures (Figure 3.8 and Figure 3.9). This might be due to the natural property of speech in the MFCC features whereas Wav_{CCCAE} was not considered as speech feature, and each feature in Wav_{CCCAE} was not strongly correlated, even though Wav_{CCCAE} was extracted from waveform. Experiments with peak detection indicated that most of the speaker-dependent M_{CCCAE} produced lesser peaks than M_{MFCC} (Figure 3.10 and Figure 3.11).

In earlier analysis (Section 3.6.2), the proposed feature, Wav_{CCCAE}, was found to have the highest CCA to the head motion. This proposed feature captures the head-motion-independent properties and speaker-dependent properties, which can effectively distinguish between two speakers in T-SNE visualisation (Figure **??** and Figure 3.6) even if they share the same head movement. In future work, this could be preferable for style transfer in motion synthesis.

3.8 Conclusion

In summary, we have proposed an approach to create a highly correlated feature with head motion from raw waveform data using CCCAE. From the objective evaluations, we can conclude that (1) CCCAE enables the creation of a more correlated feature (Wav_{CCCAE}) with head motion than Wav_{AE} and other popular spectral features such as MFCC and FBank. (2) The analysis of the features distribution among the subjects showed a clear distinct cluster for each subject in Wav_{CCCAE} only. (3) The M_{CCCAE} achieved the lowest NMSE in the test dataset, although the local CCA is not the highest. (4) The analysis based on TWV demonstrates that M_{MFCC} and M_{CCCAE} have

comparable performance. Point (3) and (4) indicate that $\text{Wav}_{\text{CCCAE}}$ is capable of being used in achieving state-of-the-art results for predicting natural head motion with the advantage of the CCCAE. (5) The movement distribution graph indicates that M_{CCCAE} tends to produce more smoothness and lesser quick movement than M_{MFCC}.

Chapter 4

LSTM-based Head Motion Estimation with DCCCAE

4.1 Introduction

In the previous chapter, we have presented DCCCAE, which effectively compresses waveform into low-dimension and highly correlated features. These compressed features with a simple FNN outperform the baseline model and M_{AE} and have a comparable performance with M_{MFCC}. In this chapter, we investigate the usage of RNN, which is better to deal with sequential data, instead of FNN. Moreover, since both input and output data in this task are sequential, we have no reason to doubt that RNN should be investigated to show the effectiveness of the proposed feature.

FNN is a layer that allows input to travel one way only: from input to output. FNN tends to be a straightforward network that associates inputs with outputs. There is no linking between any previous or next data. For our head motion estimation task, not only is the input data sequential (e.g., ASR) but also the output data (e.g., TTS). Using FNN in the regression model raises a serious consideration about whether the model has learnt the time information of the proposed feature. Another consideration could

be that since no time information is learnt by the model, the output of one data point is completely independent of the previous input. This creates a serious jerking problem when concatenating all the output data. People might argue that the input of the regression model is a stack of features and could already include temporal information, thus the necessity of RNN is not strong. However, the embedded feature is processed in a frame-wise manner and only the information which is useful to head motion is extracted by CCA objective, the time information may only be little included in the feature. Therefore, we hypothesise that RNN could do better in linking the frame-wide embedded features.

RNNs have been proven to be powerful in modelling sequential data with variable lengths. However, many types of RNN units have design flaws. One famous RNN unit is the long short-term memory (LSTM) invented by (Hochreiter and Schmidhuber [1997]). LSTM is a system that eliminates the vanishing gradient problem (Hochreiter [1991]) and prevents backpropagated errors. LSTM consists of three trainable gates to control the information flow, allowing it to learn which information should be kept and which information should be dropped. Therefore, LSTM can learn long-range dependencies.

Given the arguments above, we proposed using LSTM to replace layers of FNN in the regression model to learn time information. Instead of only modelling relationships between a set of predictor or input variables and one or more response or output variables, the model could capture time information in the proposed feature. We still kept FNN as the final output layer for functional mapping where the model has to learn how a number of input variables affect the output variable.

We tested the proposed LSTM regression model on the UoE and IEMOCAP dataset. The experimental results demonstrated that the LSTM models improved in terms of the overall performance in normalised mean square error (NMSE) and CCA metrics

and adapted the $\text{Wav}_{\text{CCCAE}}$ feature better, which makes the proposed LSTM-regression system outperform the MFCC-based system. We also designed the subjective evaluation, and the subjective results of the MUSHRA-like test indicated that the participants deemed the animations generated by models where $\text{Wav}_{\text{CCCAE}}$ was chosen to be better than the other models. The A/B test further highlighted that the LSTM-based regression model adapted better with the proposed feature $\text{Wav}_{\text{CCCAE}}$.

4.2 Related Work

We have presented the FNN-based regression method in the previous chapter. As mentioned, FNN does not consider the sequential information of the speech over time; therefore, auto-regressive (AR) model has been proposed to resolve this issue. The AR model specifies that the output variable depends linearly on its own previous values and this procedure models the temporal structure of the training data. The AR model is commonly applied in speech models, such as the HMM (Shannon and Byrne [2009];Shannon et al. [2013];). Even though both AR and RNN models can be used to model time series, the AR model has its own limitation compared to RNN. The AR model only has finite dynamic responses to time series input, whereas RNN maintains hidden layers with directed feedback connections and hence has an infinite dynamic response. In other words, RNN can fully view the time information, whereas the AR model can only view a fixed window of time information.

LSTM was introduced by Hochreiter and Schmidhuber [1997] and further investigated by Gers [2001] and it has been successfully applied in many research domains and has displayed outstanding performance related to the speech to head motion problem. Graves [2013] demonstrated the ability of LSTM networks to model long-term structure by predicting discrete text values, and by predicting the real values of handwritten trajectories. Another example by Sutskever et al. [2014] reported SOTA performance

for the language translation task.

Previous works have investigated RNN-based head motion regression with common acoustic or prosody features. Ding et al. [2015b] showed a better result applying Bi-directional LSTM (BiLSTM) trained by FBank features to their previous work, which used an FNN regression model (Ding et al. [2015a]). Unlike standard LSTM, the input of BiLSTM flows in both directions, and it's capable of utilizing information from both sides, shown in Figure 4.1. It's also a powerful tool for modeling the sequential dependencies in both directions of the sequence. However, having two layers of LSTM to deal both directions causes much slower and requires more time for training. Sadoughi and Busso [2018a] compared the FNN-based and BiLSTM-based models with prosody features. They claimed that the BLSTM-based model outperformed the FNN-based model. Furthermore, they also investigated the effectiveness of concordance correlation (CC) loss. The difference between their CC loss and our proposed CCA loss here is that they directly applied the loss in the regression, whereas we applied the CCA loss in the feature extraction. The method in our thesis might be better because the extracted feature only keeps useful information that is related to head motion and that the regression model can easily learn from.

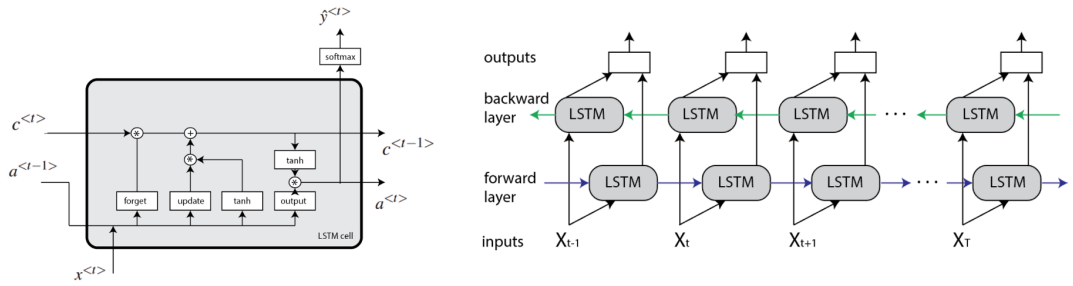


Figure 4.1: The illustration of LSTM cell (Left) and Bi-LSTM (Right).

4.3 Methodology

Figure 4.2 shows an overview of our proposed system, which can be viewed as three main modules: (1) a canonical-correlation-constrained autoencoder (CCCAE) for compressing high-dimension input into low-dimension input while sustaining correlated information between the waveform input and head motion; (2) a regression neural network for generating the head motion from the compact and correlated embedded input; (3) a neural-network-based post-filter for constructing smooth head motion from the generated output. In the training procedure, we applied mean square error (MSE) normalised by the variance of the ground truth for these three models. As models (1) and (3) are the same as the models described in Section 3.3, they are not going to be discussed again in this section. We only focus on the changes of the regression model.

4.3.1 Head Motion Regression

As discussed in the introduction and the related work, previous studies (Ding et al. [2015a], Haag and Shimodaira [2016], Greenwood et al. [2017], Sadoughi and Busso [2018b]) have conducted research using RNNs and showed better performances. A reasonable result achieved by the FNN was first proposed in our system from the experiments with $\text{Wav}_{\text{CCCAE}}$. We will further investigate the performance using LSTM with $\text{Wav}_{\text{CCCAE}}$ and how much improvement is made by LSTM compared to FNN in terms of the objective evaluation?

4.3.1.1 Feed-Forward Neural Network (FNN)

There are 7 feed-forward layers that construct the regression model here with different numbers of hidden nodes to predict head motion from the waveform embedded features, shown in Figure 4.1. The architecture and hyperparameters are the same as in the previous chapter, and we take it as the baseline model.

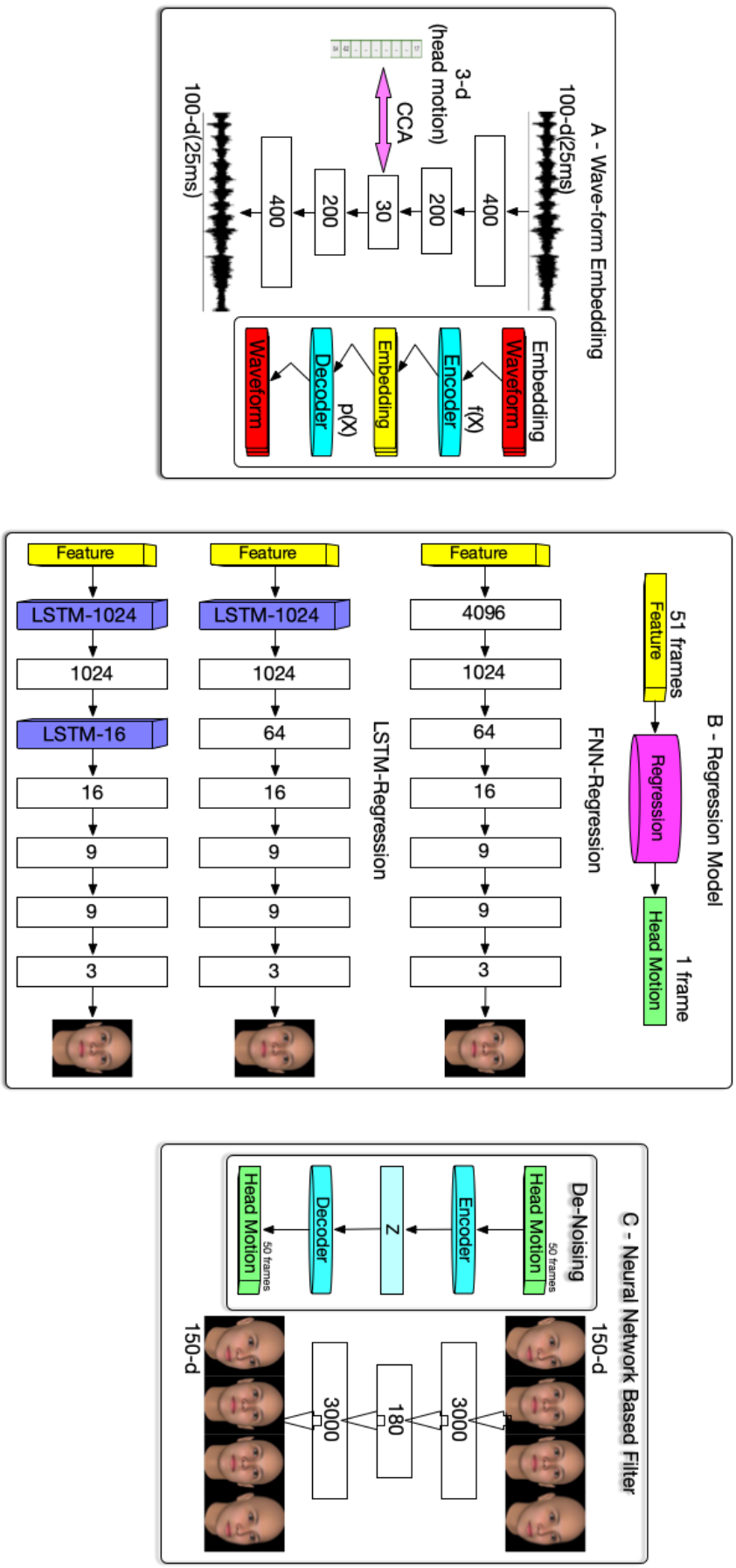


Figure 4.2: Overview of the proposed system comprised of three modules: (A) waveform embedding with CCCAE, (B) head motion regression from the features, (C) post-filter with an autoencoder. The blue LSTM in (B) indicates the differences from the previous model above.

4.3.1.2 LSTM

Different from the FNN baseline model, we have built two LSTM-based models: 1-Layer-LSTM (replaced 4096-FNN) and 2-Layer-LSTM (replaced 4096-FNN and 64-FNN). Both of the hidden cell values are much smaller than the original feed-forward layers' nodes because with the memory cell designed within LSTM, the model has a better capability to capture longer-time information. Additionally, we also did not choose to use BiLSTM, which can capture the forward and reverse time information. That is because the extracted feature, $\text{Wav}_{\text{CCCAE}}$, is compressed and highly correlated to a frame of head motion and it does not strongly cooperate with the previous and next head motion data frame. Another reason is that head motion is a type of forwarding data, it is rare to mention the movement in the backward direction. Thus, a reverse LSTM might not help in the result, but in fact it would slow the training. The following is the compact form of the equations for the forward pass of an LSTM model with a forget gate (Hochreiter and Schmidhuber [1997]), shown in Figure 4.1:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i)$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c)$$

$$h_t = o_t \circ \sigma_h(c_t)$$

where the initial values are $c_0 = 0$ and $h_0 = 0$, and the operator $\circ$ denotes the Hadamard product. The subscripts $_t$ refer to the time step.

Variables:

- $x_t \in R_d$: input vector to the LSTM unit

- $f_t \in R^h$: forget gate's activation vector
- $i_t \in R^h$: input gate's activation vector
- $o_t \in R^h$: output gate's activation vector
- $h_t \in R^h$: output vector of the LSTM unit
- $c_t \in R^h$: cell state vector
- $W \in R^{h*d}, U \in R^{h*h}$ and $b \in R^h$:weight matrices and bias vector parameters that need to be learned during training
- σ_g: sigmoid function
- σ_c: hyperbolic tangent function
- σ_h: hyperbolic tangent function

4.4 Experimental Setup

This RNN investigation task was evaluated with one male speaker (Subject A) and one female speaker (Subject D) selected from the UoE dataset. Furthermore, the same speaker used in the FNN task was selected from the IEMOCAP dataset agian. Moreover, the data selection and feature processing are the same as described in Section 3.4.

In training, we only used the frames where the target speaker for head-motion prediction was speaking so that the models learnt the relationship between speech and head motion properly. In evaluation, we made use of all the input audio sequences to generate head motion parameters.

The following notations are used in the rest experiments:

- Wav_{AE}: Embedded features extracted from waveform with the standard autoen-

coder, obj_{AE}

- $\text{Wav}_{\text{CCCAE}}$: Embedded features extracted from waveform with the proposed CC-CAE, $\text{obj}_{\text{CCCAE}}$ with $\alpha = 1$
- M_{XX}: FNN-regression model trained with XX feature
- MR_{XX}: Regression model with 1-Layer-LSTM trained with XX feature
- MR2_{XX}: Regression model with 2-Layers-LSTM trained with XX feature

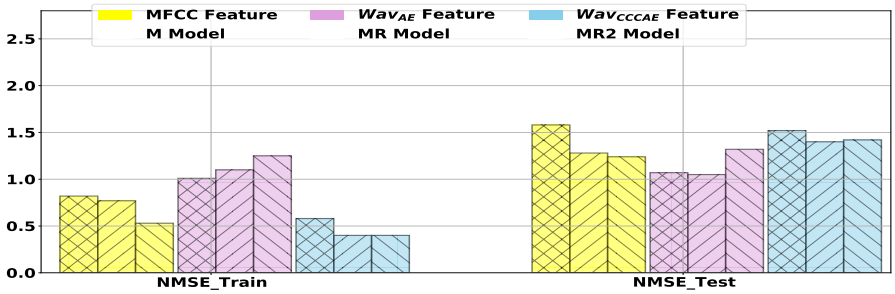
M_{MFCC}, M_{AE}, and M_{CCCAE} use the same architecture in Figure 5.5(B) to predict head motion, while each model takes different feature vectors as input.

Training was conducted on a GPU machine and a multi-CPU machine with Tensorflow version 1.12 by mini-batch training using Adam optimisation (learning rate 0.0002) (Kingma and Ba [2015]). We also employed layer-wide pre-training (Takaki and Yamagishi [2016]).

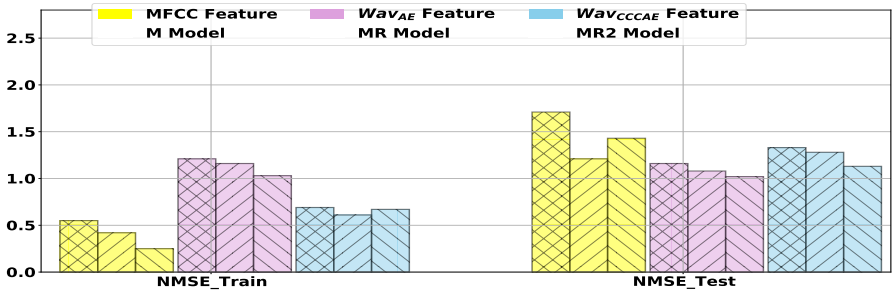
In the evaluation, test data of the same speaker is fed to the trained regression model and head motion is predicted frame by frame. After that, the output of the prediction model is then joined to form distinct head motion of 50 time frames, which are fed to the post-filtering autoencoder. The final output for animation was generated with the overlap-add method.

4.5 Results and Discussion

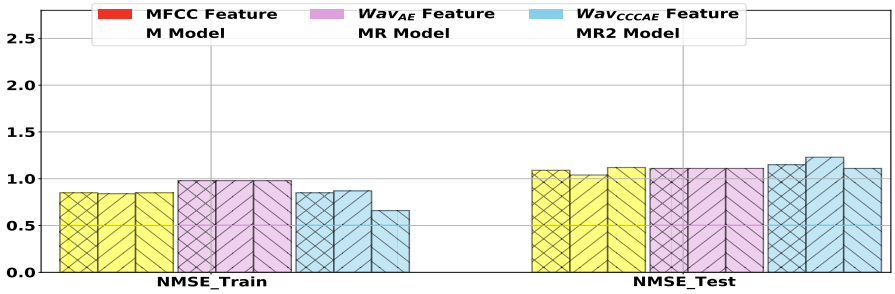
There were two main evaluations in this RNN investigation task. One set of evaluations was to calculate the similarity between two sequences of vectors, and we employed a normalised mean-squared error (NMSE), where MSE is normalised by the variance of ground truth, and local canonical correlation analysis (local CCA) (Haag and Shimodaira [2016]). Another set of evaluations was performed for subjective evaluations. In this subjective evaluation, we selected all the trained models, including LSTM-based



(a) Subject A



(b) Subject D

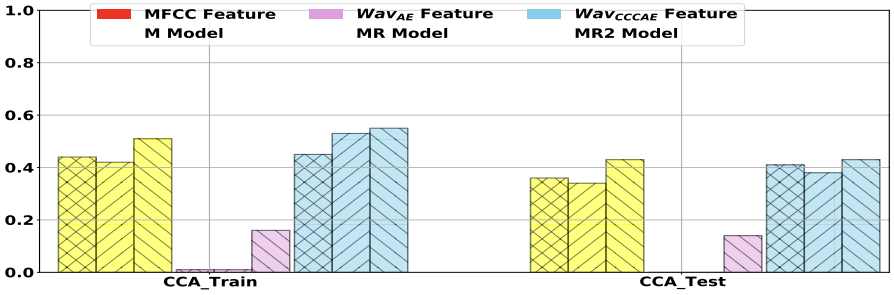


(c) IEMOCAP

Figure 4.3: Comparison of FNN and LSTM systems in terms of performance of head motion prediction, where NMSE is calculated between predicted head motion and ground truth. M: FNN model, MR: 1-Layer-LSTM that replaces 4096-FNN, MR2: 2-Layers-LSTM that replaces 4096-FNN and 64-FNN

(a) Subject A

(b) Subject D



(c) IEMOCAP

Figure 4.4: Comparison of FNN and LSTM systems in terms of performance of head motion prediction, where local CCA is calculated between predicted head motion and ground truth. M: FNN model, MR: 1-Layer-LSTM that replaces 4096-FNN, MR2: 2-Layers-LSTM that replaces 4096-FNN and 64-FNN

and FNN-based models, to be evaluated in the MUSHRA-like test and A/B test. Then, it became possible to distinguish the best model for the proposed feature.

4.5.1 Head Motion Estimation Results

In this experiment, we have done NMSE and local CCA objectives to compare the performance over FNN and LSTM models with different features. Figure 4.3 demonstrates the NMSE objective and Figure 4.4 shows the local CCA objective. A lower NMSE / higher CCA indicates better performance for the models. MR and MR2 models have better results than M models in MFCC and $\text{Wav}_{\text{CCCAE}}$ because the NMSE values have decreased (shown in Figure 4.3) and CCA values have increased respectively (shown in Figure 4.4) after switching LSTMs in Subject A and Subject D. Results for the models with Wav_{AE} reflect that there is not much difference between the models switching to LSTM and those that do not. A reason for this could be that Wav_{AE} is a low-correlated feature so, even with the advantage of LSTM, the model hardly maps the acoustics features to the head motions. Moreover, the MR and MR2 models demonstrate better adaption for highly correlated features because MR_{CCCAE} and $\text{MR2}_{\text{CCCAE}}$ outperform M_{MFCC} in CCA for Subject A and Subject D. However, there is not an obvious improvement for the benchmark dataset, IEMOCAP. The results remain mostly the same in both metrics for the test set regardless of the differences in the models and feature inputs. There is a possible reason for this is that since this dataset is recorded in a script-based manner (Busso et al. [2005]), which means the speakers are asked to speak according to the scripts. This results that there is a limited correlation between speech and head motion as the speaker mostly acts for nodding only.

4.5.2 Subjective Evaluation

Objective evaluation only shows the numerical differences between the ground truth and the generated head motion, whereas subjective evaluation reflects the opinions of

the human observers on whether the generated motion is a close match to human-likeness. Compared to our previous work (Lu and Shimodaira [2020]), which only evaluated the performance with the criteria of naturalness, we validated our models' (both FNN and LSTM models trained with three selected features respectively (M_{AE}, M_{CCCAE}, M_{MFCC}, $MR2_{AE}$, $MR2_{CCCAE}$, $MR2_{MFCC}$)) performance in the following subjective studies. We evaluated our models in two regards:

- Appropriateness - This study mainly focused on the correlation between the speech audio and the animated motion by asking the participants 'How appropriate are the head motions for the speech?'

- Model Assessment - This study asked participants to select 'Which of the following head motions are the most natural?', intending to investigate which model architecture generates the most natural head movement using the same input features.

Jonell et al. [2020] indicate that we can trust the online platforms, as there is no difference between the in-lad and the Prolific platforms in terms of the perceptual evaluation results. Therefore, we conducted our evaluation over an online platform entirely.

A group of 50 participants was recruited for this work to ensure the reliability, and they were recruited through the crowdsourcing platform Prolific, restricted to a set of English-speaking countries and native speakers only. For this evaluation, the participants were asked to evaluate both studies. The visualizing head motion software is provided by my project supervisor, Dr Hiroshi Shimodaira. The input of this software is the rotation of XYZ in radian. The output of the animation is the virtual agent head with static facial expressions (shown in Fig 4.5). One of the reasons for not covering the facial expressions or providing simple facial expression movements is that either of the ways might attract the attention of the participants and then affect the evaluation

scores. Video samples of the animation are available on the web[1].

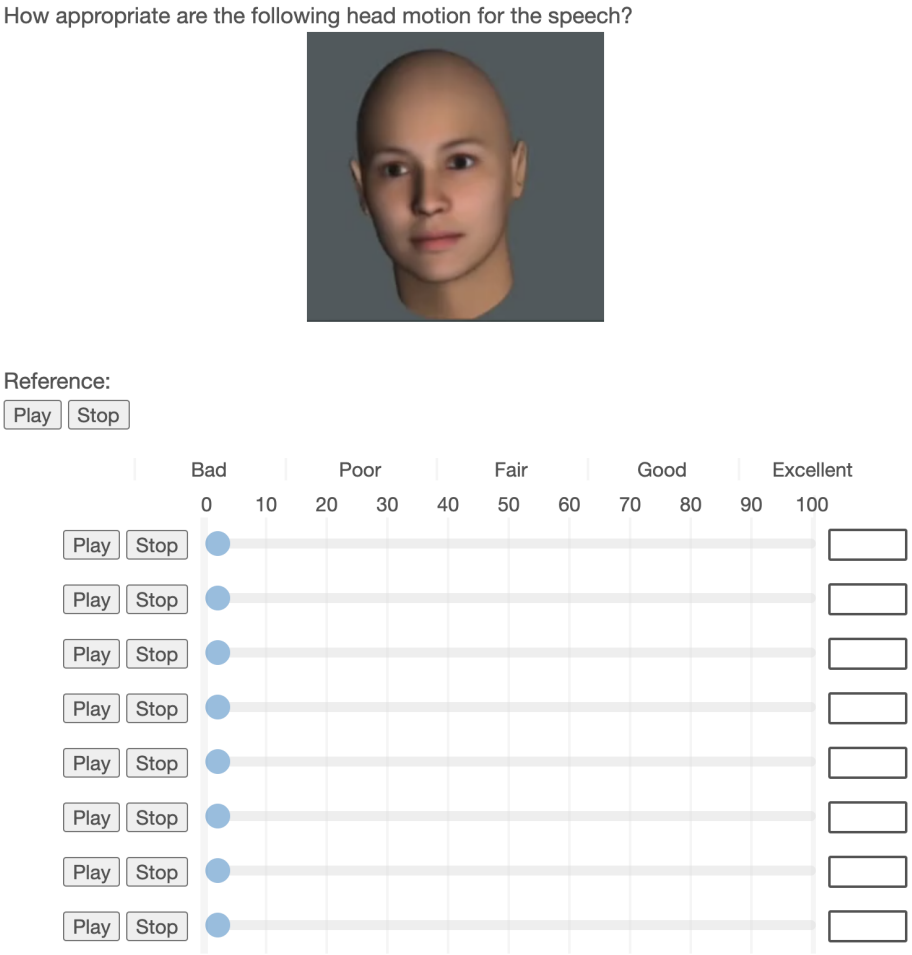


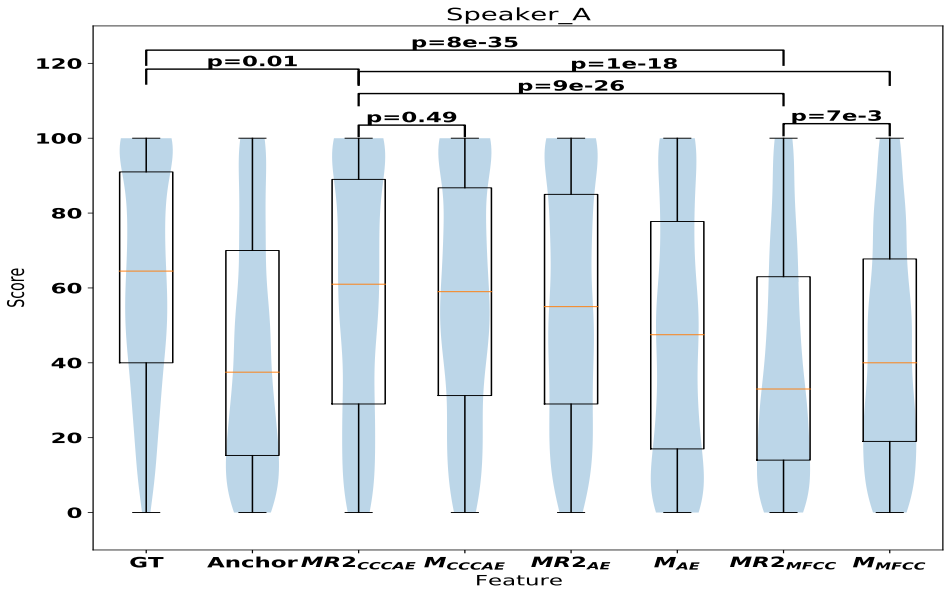
Figure 4.5: A screenshot of a MUSHRA question from the evaluation interface. Each animation was generated with the same audio utterances but different input features and model architecture. A reference video was provided and the other 8 models (GT, Anchor, M_{AE}, M_{CCCAE}, M_{MFCC}, $MR2_{AE}$, $MR2_{CCCAE}$, $MR2_{MFCC}$) were randomly shuffled and participants were asked to watch individually and give a score.

How appropriate are the head motions for the speech? A perceptual test was carried out using a similar method to MUSHRA (International Telecommunication Union). Compared to the mean opinion score (MOS) test, MUSHRA is able to obtain a better quality of scores with a minimal number of participants. We created the head motion animations with the randomly selected audio samples in the test set using eight models: ground truth (GT), anchor and both FNN and LSTM models trained with three selected

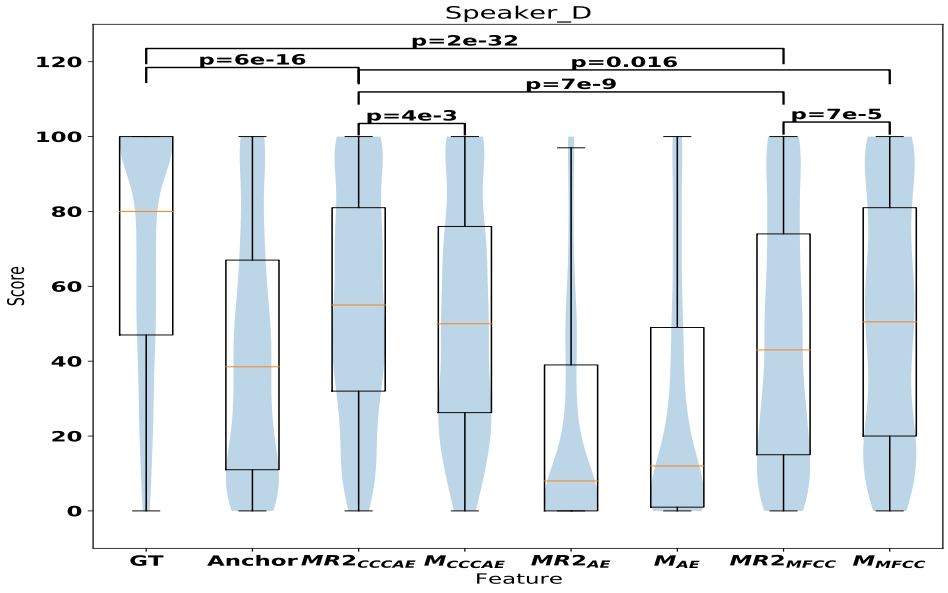
[1] https://homepages.inf.ed.ac.uk/s1569197/phd_project_demo/

features respectively (M_{AE}, M_{CCCAE}, M_{MFCC}, $MR2_{AE}$, $MR2_{CCCAE}$, $MR2_{MFCC}$). A total of 10 audio samples from each Subject A and Subject D is selected (160 animations are generated in total), and each animation lasts $8-12$ seconds long. The anchor in MUSHRA is to calibrate the scale of the scores, where the minor artifacts are not badly penalized. The anchor is created to select a different stream of head motion from another speaker with different utterances, where the resulting anchor animations are natural in terms of head motion but unsynchronised with the audio. Furthermore, a reference animation was provided as well, but it was generated with a different audio utterance than the evaluated one. This reference video was used to inform the participants how to recognise what an appropriate head motion associated with speech audio looks like. The evaluation was performed so that each participant was assigned 10 test questions, and the animations of each test question were shuffled so as to be displayed in a random order (Figure 4.5). Each participant was then requested to watch each animation carefully, and give a score between 0 and 100 for each animation. Compared to the original MUSHRA, we did not force the participants to rate the anchor to be the worst one or the ground truth to be the best. We requested that participants score at least one of the animations with the value of 100 to indicate that it is the 'ground truth'. Moreover, an attention check was incorporated in the test questions for each participant, which involved displaying a text sentence in the video such as 'Please rate this video XX'. This 'XX' would be a specific number between 10 and 100, and the participant would have to set the corresponding slider to the requested value in order to get through the attention check.

The results are displayed in Figure 4.6. From both subjects, we can initially observe that GT scored the highest, and the anchor scored about 38. This indicates that the participants were able to consistently determine the most synchronicity and the non-synchronicity between the head motion and audio. Our proposed models with Wav_{CCCAE} achieved the second highest score compared to the MFCC models and



(a) Subject A's appropriateness score



(b) Subject D's appropriateness score

Figure 4.6: The Boxplot of the MUSHRA score for both subjects' animation of each model - horizontal line indicates the median with confidence interval. The values between a pair of systems are the P-value to indicate the statistical significance.

Wav_{AE} models. The participants had different opinions on the performance between the MFCC models and Wav_{AE} models for Subject A and Subject D, respectively.

The head motion generated from MFCC achieved a better score in the objective evaluation, but a lower score in the subjective evaluation than the head motion generated from $\text{Wav}_{\text{CCCAE}}$. A possible reason for this is that while the subject is listening, MFCC is a spectral feature and does not represent non-speech information on the absolute magnitude spectrum after filter extraction and log operation, whereas waveforms are well presented. This affects models with MFCC predicting active head motion, whereas models with $\text{Wav}_{\text{CCCAE}}$ produce minor head movements while listening. An example

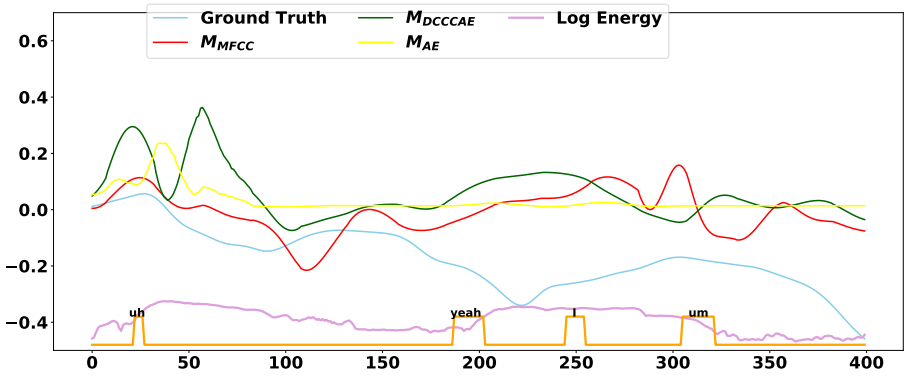


Figure 4.7: An example of trajectory-Y generated from different models. The square wave at the bottom indicates whether the subject is speaking (Up) or listening (Down). The text above the square wave is the corresponding transcript.

is shown in Figure 4.7, demonstrating noise in the non-speech region shown from the Log Energy curve, and the active head motion is generated by M_{MFCC}. Another observation from Figure 4.7 is that there is a minor head motion in the ground truth, but not in our proposed model for the non-speech region. This is another reason why the objective evaluation of M_{CCCAE} showed that it performed worse than M_{MFCC}. Participants may have felt that active head motion went against natural human instincts while listening. Even though the GT showed animated head motion in the listening region as well, participants still preferred the ground truth over models with $\text{Wav}_{\text{CCCAE}}$, which

indicates that the head motion generated by MFCC is unnatural. This also suggests that an objective approach is quantifiable, whereas subjective approaches are open to greater interpretation based on personal feeling (Leahu et al. [2008]).

We also applied the significance test (paired t-test) to the mean score distributions across different pairs of the models. We made comparisons from three perspectives: 1) whether the GT motion is significantly different from the predicted ones (GT VS $MR2_{CCCAE}$ and GT VS $MR2_{MFCC}$); 2) whether the LSTM model is significantly different from the FNN model ($MR2_{CCCAE}$ VS M_{CCCAE} and $MR2_{MFCC}$ VS M_{MFCC}); 3) whether Wav_{CCCAE} is better than MFCCs ($MR2_{CCCAE}$ VS M_{MFCC} and $MR2_{CCCAE}$ VS $MR2_{MFCC}$).

According to the results of the significance test shown in Figure 4.6, we answer the three perspectives accordingly. (1) The GT motion is not significantly different from $MR2_{CCCAE}$ but is significantly different from $MR2_{MFCC}$ in Subject A. Yet, the GT motion is significantly different from both models in Subject D. (2) For MFCC models, it is significantly different between LSTM and FNN models. However, for Wav_{CCCAE} it is not significantly different in Subject A, but it is significantly different in Subject D between LSTM and FNN. (3) $MR2_{CCCAE}$ is significantly different from the models trained with MFCCs in Subject A and $MR2_{MFCC}$ in Subject D, but not for M_{MFCC} in Subject D. In summary, $MR2_{CCCAE}$ significantly outperformed the models trained with MFCCs, and the difference between GT and $MR2_{CCCAE}$ is not statistically significant in Subject A. However, $MR2_{CCCAE}$ is only comparable to the models trained with MFCCs and is worse than GT in Subject B. Last, the difference between the LSTM models and the FNN models in both subjects is not statistically significant. This implies that their performances are comparable.

Which of the following head motions is the most natural? We conducted this second

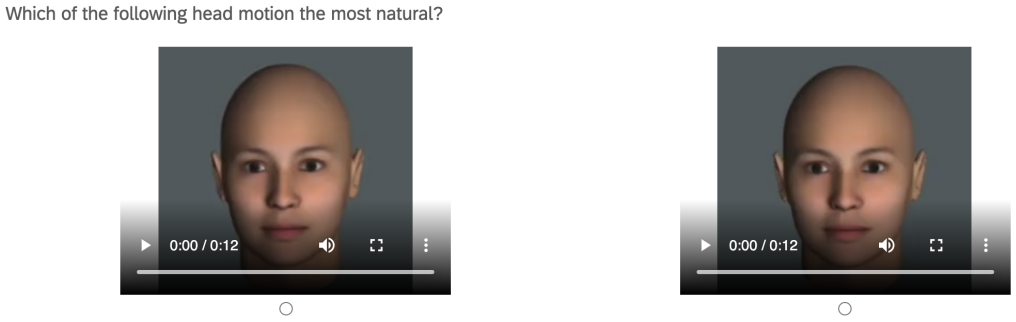


Figure 4.8: A screenshot of the A/B test from the evaluation interface. Both animations were generated with the same input feature, but different in the model architecture. Right: LSTM, Left: FNN.

study using an A/B test to ask the participants to simply select which head motion video is more natural than the other (Figure 4.8). Our intention was to compare the feed-forward neural network and recurrent neural network with the same input features. These videos are generated from Wav_{AE}, $\text{Wav}_{\text{CCCAE}}$ and MFCC despite the fact that the models with Wav_{AE} only produces minor head motions.

As shown in Figure 4.9, according to the participants, $\text{MR2}_{\text{CCCAE}}$ was always better than M_{CCCAE}, whereas they had different opinions regarding the Wav_{AE} and MFCC models in both subjects. A possible reason for different opinions on both subjects under Wav_{AE} is that since both model architectures produce minor head motions, participants might just randomly prefer a model. Then for MFCC models, as discussed above, models with MFCC always predict active head motion in both speaking and listening status. Participants could be confused and thus could have found it difficult to pick the better-performing one. This conclusion could be drawn from the fact that both result bars under Wav_{AE} and MFCC were similar and were nearly 10% away from the borderline. From the results, we also observe that LSTM always performed better in Subject A and generated more preferable head motions with all the features according to the participants. Lastly, the results of the proposed features $\text{Wav}_{\text{CCCAE}}$ were consistent in both subjective studies as the LSTM was better than FNN for $\text{Wav}_{\text{CCCAE}}$ for

both subjects.

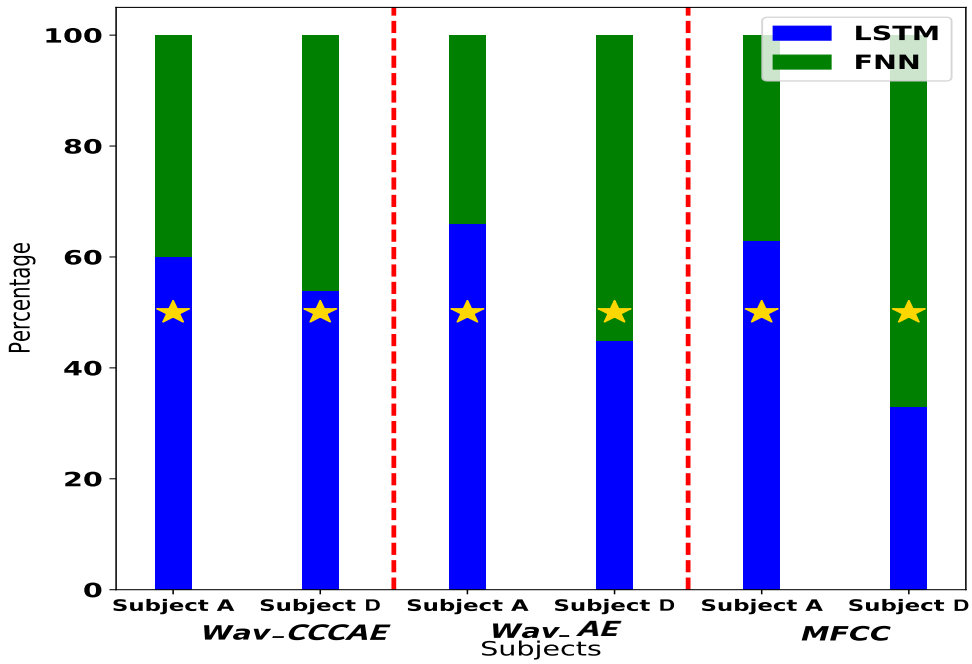


Figure 4.9: The barplot of the A/B test for different model architectures. The star position indicates the 50% border line.

4.6 Further Considerations

The proposed MR models outperformed M models and adapted the proposed feature $\text{Wav}_{\text{CCCAE}}$ better as MR_{CCCAE} achieved better results in both objective and subjective evaluations compared to the models trained with MFCC. However, nowadays, the LSTM model is not the SOTA model anymore. Many more advanced models were developed since the period of this research. To better prove the effectiveness of our proposed feature and system, we must explore more architectures in the future.

Another important point to be considered is that instead of exploring RNN in the regression model, we should also consider exploring it in the feature extraction part. That is because the input feature and output feature is waveform and the model con-

siders stacking correlation, which means we stack time information while maximising the correlation between the stacked 51-frame extracted feature and the motion feature. Moreover, it would also be interesting to apply CCA objectives in the regression model. This could further correlate the predicted motion and the ground truth motion instead of applying the reconstruction error only.

4.7 Conclusion

We proposed LSTM-based models replacing the FNN-based regression model, benefitting from the long sequence-to-sequence adaptation in dealing streaming data. The LSTM-based models learnt to produce better head motion in practice. The LSTM-based regression models were able to boost the overall performance in NMSE and CCA, adapted better with the proposed feature ($\text{Wav}_{\text{CCCAE}}$) than MFCC; in this regard, MUSHRA test results suggest that the participants deemed the animations generated by models with $\text{Wav}_{\text{CCCAE}}$ to be better than other models. The A/B test further highlighted that the LSTM-based regression model adapted better with the proposed feature $\text{Wav}_{\text{CCCAE}}$.

In conclusion, the improvement of the objective evaluation and the outstanding subjective results suggest that with the help of the CCCAE, $\text{Wav}_{\text{CCCAE}}$ has the potential to be one of the task-specific features for generating head motion, achieving SOTA results.

Chapter 5

Upper Body Motion Estimation using Double-DCCCAE

5.1 Introduction

When people converse, many motions (gesture, body and head movments) occur spontaneously (Hadar et al. [1983], Mcneill [1994]). These motions are transmitted as non-verbal signals to the listeners and help the listeners better understanding what is being expressed (Knapp et al. [2013], Matsumoto et al. [2013]). As such, non-verbal motion is a key factor for the conversational agents or social robots to interact with us and act human-like (Breazeal et al. [2005], Salem et al. [2013]).

As discussed in Chapter 1, non-verbal motion consists of gesture, lip, head and body movements and each type of motion plays a different role within the conversation. In the previous chapters (Chapter 3 and Chapter 4), we investigated the effectiveness of the $\text{Wav}_{\text{CCCAE}}$ in estimating head motions and found that DCCCAE enables the creation of a highly correlated and low-dimension feature. In this chapter, instead of estimating head motion only, we study the estimation of body, gesture and head motion. In other words, we would like to investigate the usage of DCCCAE for estimating

upper body motion. Before discussing how the estimation process changes in this regard compared to the head motion task, we observe a scenario that researchers tend to use multiple frames of speech to estimate one frame of motion. We find a possible explanation could be that a complete head motion is measured to be last at least 400 ms (Hofer and Shimodaira [2007]), and the same should apply to the other types of motion. Moreover, since the speaking rate of a human is 2.3 words per second, to fully make usage of the context information in the conversation, the model requires much more speech frames in the input stage.

This observation leads to a question: since motions are continuative, then when we want to estimate multiple frames of motion at once, are we required to input stacked speech information? The literature demonstrates that the complexity of the model increases if we would like to generate multiple frames of motion (Kucherenko et al. [2019], Lu and Shimodaira [2020]). The hardware limitation does not allow us to perform such experiments; Additionally, the correlation between multiple frames of speech information and a frame of body motion is not strong, not to mention that the correlation becomes weaker after stacking blocks of multiple speech information to generate multiple frames of body motion. The results of the experiment conducted in this study also shows that the correlation becomes weaker. RNN may be a reasonable solution for the sequence-to-sequence data estimation/prediction. However, we cannot ignore the weakness of the RNN for long time-step data in terms of gradient vanishing and exploding. To resolve these problems, we proposed a double deep canonical-correlation-constrained autoencoder (D-DCCCAE), a frame-based system that can estimate temporal sequence.

Our proposed system consists of three parts: the D-DCCCAE, a frame-based regression model and a post-filter. The auto-encoders are used to compress the information of the sequential data (e.g., speech information or body motion), as well as maintain possible higher correlations with other sequential data. The frame-based regression

predicts the sequential motion embedding in a frame-by-frame manner from the wave embedding. The predicted frame-based motion embedding is further decoded by the trained decoder and interpreted as the sequential body motion movements. Last, we apply an NN-based filter to smooth the generated movements.

The features obtained with the proposed approach are more highly correlated compared to the correlation between raw waveform and MFCC with the motion data. We submitted our model to the GENEA2020 challenge (Kucherenko et al. [2020]) and evaluated it with other participants' models and baseline models in a subjective test.

5.2 Related Work

Human gesticulation is highly complex and still not fully understood, even though much research has been done investigating it. The high-level vocabulary (agent, gesture and speech) and more fine-grained terminology about gesture properties, defined in the literature, are reviewed in Section 5.2.1. Nevertheless, there is a consensus that speech and gestures correlate strongly. Hence there has been much work analyzing human gesticulation with respect to speech timing and content, which is reviewed in Section 5.2.2.

5.2.1 Body Motion Concepts

Generating co-speech gestures has been an essential task in Human-Agent Interaction for several decades. As the development goes on, there are several terms that are defined along the way. Thus, we have to understand these terms before we can get into details. The following sections would split into two: first, we are going to review the gesture properties with respect to gesture functionalities, dimensions and gesture phrases and phases. Second, we would define the representation we used in this thesis.

5.2.1.1 Gesture Properties

Several gesture properties are essential in gesture research and are outlined in the following section.

Gesture functionalities

Several theories suggest how and why gestures occur during communication and thinking. Mechanistic theories mostly propose how gestures arise during communication and thinking (Mcneill [1994], McNeill [2005], Hostetter and Alibali [2008, 2018]). Functionalist theories, on the other hand, try to explain why we use gestures and the functions that gestures serve during communication and thinking, both for the speaker and the listener (Goldin-Meadow et al. [2001], Kita and Özyürek [2003], Pouw et al. [2014], Church et al. [2017], Kita et al. [2017], Novack and Goldin-Meadow [2017]).

Özer and Göksun [2020] had summarised the functionality of the gesture in the following two ways.

- Gestures affect communication between interlocutors.
- Gestures affect speakers' and listeners' cognitive processes.

Speakers and listeners employ gestures for communicative purposes. Speakers produce gestures to communicate information, and listeners, in turn, benefit from these gestures to comprehend the to-be-communicated message. In such, gestures help activate, maintain, manipulate, and package visual, spatial, and motoric information for speaking and thinking. Gestures reduce cognitive load by keeping spatial-motoric information active in working memory and by projecting internal representations to an external space(Pouw et al. [2014]).

Functional gesture theories assert that gestures help to convey information during communication and manage cognitive load during speaking, thinking, and learning (Kita et al. [2017], Novack and Goldin-Meadow [2017]). This suggests that gesture use

and processing are sensitive to the cognitive dispositions of the speakers and listeners. People might convey gestures to manage and compensate for their limited cognitive resources.

Gesture dimensions

There are different gesture properties that have been defined over the decades, one of the most commonly used in gesture research is defined by McNeill [1992, 2005], who distinguished the following gesture dimensions depending on their function:

- Beat gestures are used for emphasis and usually correlate with the speech prosody (e.g., intonation and loudness).
- Deictic gestures create a reference, generally by pointing to an object or orientation in space. They can be abstract as well as concrete.
- Iconic gestures represent some aspect of the scene being described in speech, such as the shape or size of an object.
- Metaphoric gestures represent an abstract concept that is not physically present.

McNeill [2005] suggests differentiating gestures in terms of dimensions rather than disjunctive categories since several of the dimensions could be activated at the same time. Thus, a given gesture could be, for example, both iconic and deistic or both beat and metaphoric.

The last three gesture dimensions, sometimes referred to as representational gestures, depend on the content of the speech - its semantics - while the first dimension instead depends on the audio signal - the acoustics.

Gesture phrases and phases

Kendon [1981] analyzed the structure of how gesticulation unfolds across time. He saw that gesticulation could be split into gesture units - intervals starting when the hands ”begin to depart from a position of relaxation until the moment when they finally return

to one." Gesture units can consist of multiple gestures. Each gesture within a gesture unit is also called a gesture phrase. Putting it simply: gesture phrases are separate gesture units that are sequences of gestures that start and end in rest positions of the limbs.

Each gesture phrase can be further split into gesture phases: preparation, stroke, hold and retraction, using the terms from McNeill [2005]. (as shown in Figure 5.1). During the preparation phase, the hands leave the rest pose and get into the gesture's starting position. An example would be raising the hands to a needed height. Hold phase means holding the hands in a fixed position; this phase can happen both before and after the stroke. The stroke is the expressive phase of the gesture in which "the meaning of the gesture is expressed" (McNeill [1992]). The stroke is characterized by a "distinct peaking of effort" (Kendon [1981]). Finally, during the retraction (also called relaxation) phase, hands are brought back into the rest pose. All phases are optional except for the stroke, which is the expressive phase of the gesture.

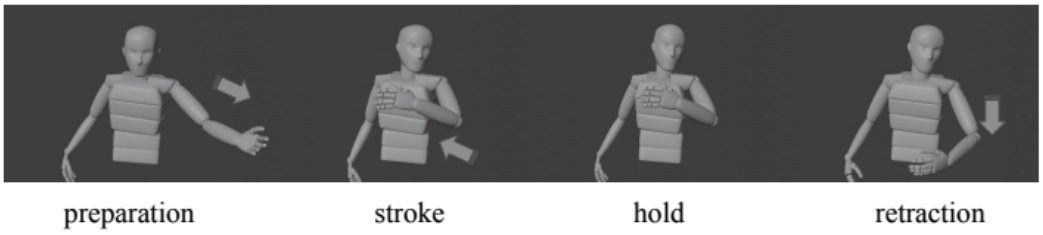


Figure 5.1: Illustration of the gesture phases.

5.2.1.2 Gesture Representation

Table 5.1 outlines many of the motion capture formats in use today along with URLs for additional formatting information.

For the remainder of this section, the BVH file formats are examined in more detail, which includes an explanation of the formatting of the file and the processes needed in order to correctly display a given animation. BVH formats have been selected for

File Extension	Associated Company / Description	File Format Reference
ASC	Ascension	No Link
ASF & AMC	Acclaim	http://www.darwin3d.com/gamedev/acclaim.zip
ASK & SDL	BioVision/Alias	No Link
BVA & BVH	BioVision	http://www.biovision.com/bvh.html
BRD	LambSoft Magnetic Format	http//www.dcs.shef.ac.u,/∼mikem/fileformats/brd.html
C3D	Biomechanics, Animation and Gait Analysis	http://www.c3d.org/c3d_format.htm
CSM	3D Studio Max, Character Studio	http://www.dcs.shef.ac.uk/∼mikem/fileformats/csm.html
DAT	Polhemous	No Link
GTR, HTR & TRC	Motion Analysis	http://www.cs.wisc.edu/graphics/Courses/ cs-838-1999/Jeff{HTR.html, TRC.html}
MOT & SKL	Acclaim-Motion Analysis	(Under Development - http://www.cs.wisc.edu/graphics/Courses/ cs-838-1999/Jeff/SKL-MOT.html)

Table 5.1: Motion Capture File Formats and References For Additional Format Information (Meredith and Maddock [2001])

expansion here because the BVH format is the file format provided by the GENEA 2020 and this BVH format tends to be the more common format used and a successful implementation of the decoder has been achieved. The BVH format succeeded Bio-Vision's BVA data format with the noticeable addition of a hierarchical data structure representing the bones of the skeleton. The BVH file consists of two parts where the first section details the hierarchy and initial pose of the skeleton and the second section describes the channel data for each frame, thus the motion section. Illustrations of the base position and the first frame of an animation are given in the following Figure 5.2, where the data is listed in Figure 5.2. The example BVH file in Figure 5.2 will be used to further discuss the BVH file format in the remainder of this section.

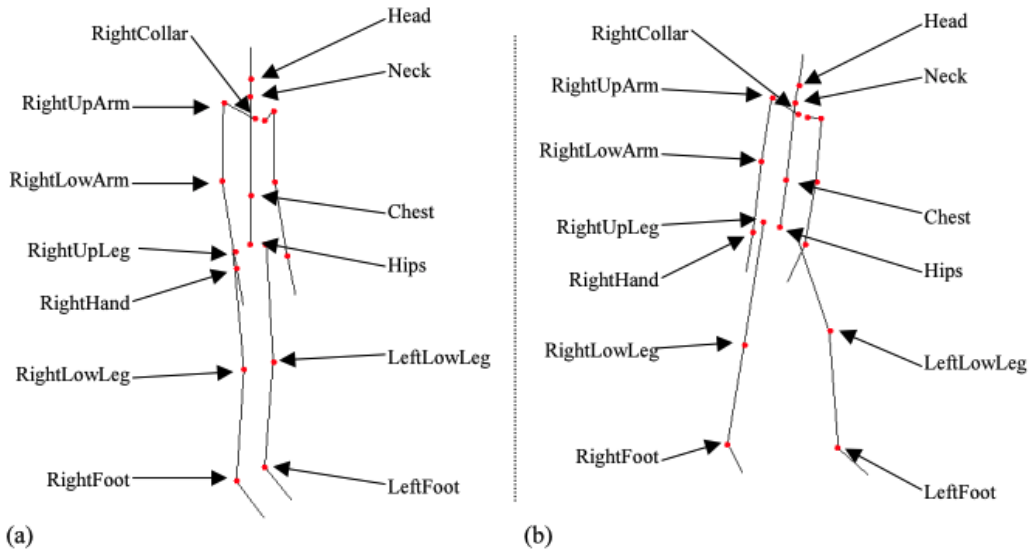


Figure 5.2: Skeletal Structure of the sample BVH file; (a) base position; (b) first frame of the animation

The hierarchical section of the file starts with the keyword *HIERARCHY*, which is followed on the next line by the keyword *ROOT* and the name of the bone that is the root of the skeletal hierarchy. The *ROOT* keyword indicates the start of a new skeletal hierarchical structure and although the BVH file is capable of containing many skeletons, it is usual to have only a single skeleton defined per file.

TestSeq001.bvh

```
HIERARCHY
ROOT Hips
{
    OFFSET -14.64140 90.27770 -84.91600
    CHANNELS 6 Xposition Yposition Zposition Zrotation Xrotation Yrotation
    JOINT Spine
    {
        OFFSET 0.00000 13.20850 -1.60436
        CHANNELS 3 Zrotation Xrotation Yrotation
        JOINT Spine1
        {
            OFFSET 0.00000 8.61716 0.00000
            CHANNELS 3 Zrotation Xrotation Yrotation
            JOINT Spine2
            {
                OFFSET 0.00000 8.61717 0.00000
                CHANNELS 3 Zrotation Xrotation Yrotation
                JOINT Spine3
                {
                    OFFSET 0.00000 8.61717 0.00000
                    CHANNELS 3 Zrotation Xrotation Yrotation
                    JOINT Neck
                    {
                        OFFSET 0.00000 11.07920 1.10792
                        CHANNELS 3 Zrotation Xrotation Yrotation
                        JOINT Neck1
                        {
                            OFFSET 0.00000 7.08032 0.00000
                            CHANNELS 3 Zrotation Xrotation Yrotation
                            JOINT Head
                            {
                                OFFSET 0.00000 7.08031 0.00000
                                CHANNELS 3 Zrotation Xrotation Yrotation
                                End site
                                {
                                    OFFSET 0.00000 0.00000 0.00000
                                }
                            }
                        }
                    }
                    JOINT RightShoulder
                    {
                        OFFSET -0.01000 7.91373 5.19711
                        CHANNELS 3 Zrotation Xrotation Yrotation
                        JOINT RightArm
                        {
                            OFFSET -18.41580 0.00000 0.00000
                            CHANNELS 3 Zrotation Xrotation Yrotation
                            JOINT RightForeArm
                            {
                                OFFSET -29.11090 0.00000 0.00000
                                CHANNELS 3 Zrotation Xrotation Yrotation
                                JOINT RightHand
```

TestSeq001.bvh

```
        JOINT RightHandIndex2
        {
            OFFSET -5.48836 0.00000 0.00000
            CHANNELS 3 Zrotation Xrotation Yrotation
            JOINT RightHandIndex3
            {
                OFFSET -3.01859 0.00000 0.00000
                CHANNELS 3 Zrotation Xrotation Yrotation
                End site
                {
                    OFFSET 0.00000 0.00000 0.00000
                }
            }
        }
    }
    JOINT RightHandMiddle1
    {
        OFFSET -14.26970 0.00000 -0.09147
        CHANNELS 3 Zrotation Xrotation Yrotation
        JOINT RightHandMiddle2
        {
            OFFSET -6.17441 0.00000 0.00000
            CHANNELS 3 Zrotation Xrotation Yrotation
            JOINT RightHandMiddle3
            {
                OFFSET -3.56743 0.00000 0.00000
                CHANNELS 3 Zrotation Xrotation Yrotation
                End site
                {
                    OFFSET 0.00000 0.00000 0.00000
                }
            }
        }
    }
    JOINT RightHandRing1
    {
        OFFSET -12.69180 0.00000 -3.06433
        CHANNELS 3 Zrotation Xrotation Yrotation
        JOINT RightHandRing2
        {
            OFFSET -5.62556 0.00000 0.00000
            CHANNELS 3 Zrotation Xrotation Yrotation
            JOINT RightHandRing3
            {
                OFFSET -3.56744 0.00000 0.00000
                CHANNELS 3 Zrotation Xrotation Yrotation
                End site
                {
                    OFFSET 0.00000 0.00000 0.00000
                }
            }
        }
    }
```

```
    JOINT LeftUpLeg
    {
        OFFSET 9.98441 0.00000 0.00000
        CHANNELS 3 Zrotation Xrotation Yrotation
        JOINT LeftLeg
        {
            OFFSET 0.00000 -42.11800 0.00000
            CHANNELS 3 Zrotation Xrotation Yrotation
            JOINT LeftFoot
            {
                OFFSET 0.00000 -45.00070 0.00000
                CHANNELS 3 Zrotation Xrotation Yrotation
                JOINT LeftForeFoot
                {
                    OFFSET 0.00000 -3.74824 0.00000
                    CHANNELS 3 Zrotation Xrotation Yrotation
                    JOINT LeftToeBase
                    {
                        OFFSET 0.00000 0.06325 14.43110
                        CHANNELS 3 Zrotation Xrotation Yrotation
                        End site
                        {
                            OFFSET 0.00000 0.00000 0.00000
                        }
                    }
                }
            }
        }
    }
}
MOTION
Frames: 2355
Frame Time: 0.050000
0.0 0.0 0.0 0.0 0.0 0.0 -10.88158647560735 6.155165503642376 1.654852079499059
-2.019510168295758 5.235086401300089 0.8192040453695822 -2.5405179033761884
-0.3180895727684533 2.2897544970739703 -2.113191653804954 2.0808800016976954
-0.30772675123927823 -2.6205095327936063 16.500390839625542 2.7623039818854975
-0.10790376439916441 -6.018582093285756 0.5268935519052542 -0.40124568204009325
-23.20062693127817 0.1273248042936706 -3.669162924686501 1.0275149095743703
-11.364991033327767 58.61287608068841 25.996585651926505 33.28486709035025
2.1534372747081023 -4.433234909722573 105.73024003236046 -5.754846256156417
1.1385483305694897 -9.711869902456687 18.0116 2.65794 8.11701 -9.93488e-06 7.38895e-08
-9.03245e-06 -5.11709e-07 -8.25489e-07 1.38866e-06 -13.9501 -4.71279e-06 8.47328e-07
-1.76648e-07 3.98573e-07 -1.13742e-06 -7.24577e-07 -1.81966e-06 1.63278e-06 -15.4545
6.04519e-06 1.31137e-05 9.68548e-06 1.63632e-06 5.07264e-06 2.42561e-07 6.13646e-07
-5.3625e-07 -3.54073 1.39824e-06 5.791e-06 2.33234e-07 5.54681e-07 -5.97039e-07 2.33234e-07
5.54681e-07 -5.97039e-07 28.4328 6.44384e-06 -3.58119e-06 2.02062e-05 2.43091e-06
-3.72353e-06 -2.51462e-06 8.84088e-08 -2.23791e-06 4.517049833906226 1.2040929162370928
14.870864928657639 -63.726910508281826 41.16948310349212 -37.523456195049874
8.747228991846182 2.9356234349670896 -102.74875760843099 -1.8123444415919598
3.390108708279821 -2.7322169948472244 -31.1647 0.16201 -0.267878 -8.50445e-08 2.28641e-07
3.50291e-07 -6.37834e-08 1.71481e-07 2.62719e-07 6.92034 1.49494e-06 -8.71045e-07
```

Figure 5.2: Example of BVH file

The remaining structure of the skeleton is defined in a recursive nature where each bone's definition, including any children, is encapsulated in curly braces, which is delimited on the previous line with the keyword *JOINT* (or *ROOT* in the case of the root bone) followed by the name of the bone. With the introduction of a left curly brace, it is good practice to indent the bone's content (with a tab) and align the closing curly brace with the corresponding opening one. The bone names identified by the prefix *JOINT* or *ROOT* are not referenced again in the file and hence redundant, however some parses (for example Character Studio R2.2) require a bone name in order to correctly parse the file. Furthermore, although the hierarchical indentation is not absolutely necessary, it does assist in making the file more readable for humans.

Within the definition of each bone, the first line, delimited by the keyword *OFFSET*, details the translation of the origin of the bone with respect to its parent's origin (or globally in the case of the root bone) along the x,y,z-axis respectively. The offset serves a further purpose of implicitly defining the length and direction of the parent's bone, however the problem with this is in defining the length and direction of a bone that has multiple children. Normally a good choice for determining the bone length in this situation is to use the first child offset definition to infer the parental bone information and treat the offset data for other child nodes simply as offset values.

The second line of a bone's definition is prefixed with the keyword *CHNNELS* which defines the DOFs for the current bone. The importance of the order that the channels are presented in two-fold. First, the order that each channel is seen in the hierarchy section of the file exactly matches the order of the data in the motion section of the file. For example, the motion section of the file contains information for the channels of the root bone in the order defined in the hierarchy, followed by the channel data for it's first child, followed by the channel data for that child and so on through the hierarchy. The second point to note with regards to the channel ordering is that the concatenation order of the Euler angles when creating the bone's rotation matrix needs to follow

the order depicted in the *CHANNEL* section. It is important to note this because the Euler order is specified for each bone, therefore it is possible to have different orders for different bones, which needs to be accounted for in order to get a correct looking animation. Figure 5.3 illustrates a segment of a BVH file in which rotational channels are specified differently for different joints.

```
OFFSET -9.98441 0.00000 0.00000
CHANNELS 3 Zrotation Xrotation Yrotation
JOINT RightLeg
{
    OFFSET 0.00000 -42.11790 0.00000
    CHANNELS 3 Zrotation Xrotation Yrotation
    JOINT RightFoot
    {
        OFFSET 0.00000 -45.00020 -0.00001
        CHANNELS 3 Zrotation Xrotation Yrotation
        JOINT RightForeFoot
        {
            OFFSET 0.00000 -3.74820 0.00000
            CHANNELS 3 Zrotation Xrotation Yrotation
            JOINT RightToeBase
            {
                OFFSET 0.00000 0.06325 14.43120
                CHANNELS 3 Zrotation Xrotation Yrotation
                End site
                {
                    OFFSET 0.00000 0.00000 0.00000
                }
            }
        }
    }
}
```

Figure 5.3: Example BVH fragment containing varying orders of joint rotations

After $OFFSET$ and $CHANNEL$ lines, the next non-nested lines in the bone definition are used to define child items, starting with the keyword $JOINT$, however in the case of end-effectors, a special tag is used, "$EndSite$", which encapsulates an $OFFSET$ triple that is used to infer the bone's length and orientation.

Once the skeletal hierarchy is defined, the second section of a BVH file, which is denoted with the keyword $MOTION$, contains the number of frames in the animation, frame rate and the channel data. The line containing the number of frames starts with the keyword "$Frames$:" which is followed by a positive decimal integer (as opposed to hexadecimal or octadecimal) that is the number of frames. The frame rate is on a line starting with "$FrameTime$:" which is followed by a positive float that represent the duration of a single frame. To convert this into a frames per second format you simply need to divide 1 by the frame time. Once the number of frames and frame time has been defined,the rest of the file contains that channel data for each bone in the order they were seen in the hierarchy definition, where each line of float values represents an animation frame.

Processing the Data

The first thing that needs to be done in order to display the motion is to determine each bone's local transform, for which the general equation was given in Equation 2.3 as $M = TRS$. Since BVH formats do not contain scaling information we only need consider the rotation and translation matrices to construct the local transform. The construction of the rotation matrix, R, can be easily done by multiplying together the rotation matrices for each of the different channel axes in the order they appeared in the hierarchy section of the file. For example, consider the following channel description for a bone:

$$CHANNELS \quad 3 \quad Zrotation \quad Xrotation \quad Yrotation \tag{5.1}$$

This would mean that the compound rotation matrix, $\mathbf{R}$, is calculated as illustrated in Equation 5.2.

$$\mathbf{R} = R_z \, R_x \, R_y \tag{5.2}$$

Once the composite rotation matrix is calculated, using a homogeneous coordinate sys-

tem, the translation components are simply the first 3 cells of the 4^{th} column (whereas the rotational components take up the top left 3×3 cells), as illustrated in Equation 5.3. (Note: If pre-multiplication of the vertices were being used, the translation components would take up the first 3 cells in the 4^{th} row.) Normally, the root is the only bone that has per-frame translation data, however each bone has a base offset that needs to be added to the local matrix stack. Therefore, T_x,T_y and T_z represent the summation of a bone's baseposition and frame translation data.

$$\mathbf{M} = \begin{bmatrix} R & R & R & T_x \\ R & R & R & T_y \\ R & R & R & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.3}$$

The global positions for each bone origin can be calculated and from the origin the bone is drawn using the offset information in the hierarchy section of the file by the following equation and the derivations of the local transforms.

$$\mathbf{M}^n_{global} = \prod_{i=0}^{n} \mathbf{M}^i_{local} \tag{5.4}$$

where n is the current bone whose parent bone is $n-1$ and $n=0$ is the bone at the root of the hierarchy.

Equation 5.5 exemplifies this process for the LeftFoot in Figure 5.3, where v'_0 and v'_1 are the endpoints of the bone whose local orientation is given by v and M_i are the local transforms of the bones involved in the hierarchical chain. The vector of the right of the first expression in Equation, $[0,0,0,1]^\top$, represents the local origin of the *RightLeg*,

which is transformed into its global position by the equation.

$$\begin{aligned} v_0^{'} &= M_{Hips}\, M_{RightLeg}\, M_{RightFoot}\, M_{RightForeFoot}\, M_{RightToeBase}\, [0,0,0,1]^{\top} \\ v_1^{'} &= M_{Hips}\, M_{RightLeg}\, M_{RightFoot}\, M_{RightForeFoot}\, M_{RightToeBase}\, v \end{aligned} \tag{5.5}$$

During playback of animations that are in a hierarchical format, if the motion is to be used multiple times and unchanged then to increase performance the vertices can be calculated once and then stored for later cycles. However, if real-time modifications are to be performed on the motion then keeping the data in a hierarchical format greatly increases the ease with which the character posture can be edited. Therefore pre-calculating absolute vertex positions of bones provide no advantage over a hierarchical rending algorithm, in fact, could even result in a loss of performance. This is because calculating the positions on the fly results in the global transformation being cached as opposed to pre-calculating the values, storing them and then recalling them from primary memory, which requires more instruction commands and additional access to slower memory compared to cache memory.

While this hierarchical data structure may assist in the orientations of bones and the skeleton as a whole, the computation load required to display the skeleton is far from efficient. This is because each branch at each level in the hierarchy requires an extra matrix multiplication as outlined in Equation 5.5, which in turn is made up of multiple transformation matrices. In order to improve efficiency, the local transforms can be pre-compiled into a single matrix that is ready for stack multiplication, and for optimal performance, all of the bone end-points could be pre-calculated using a variant of Equation 5.5. This would result in simply pushing the absolute vertex positions into the graphics pipeline, however, this optimal rendering format means that it is virtually impossible to modify the existing motion with any meaningful results because all of the hierarchical information has been lost.

There are a number of problems inherent in the BVH file format. Most noticeable is the fact that there is no explicit bone orientation. Although the bone lengths can be inferred from child's bones, the problem comes with multiple children, as previously discussed - which child do you use to infer the parent's bone length? Furthermore, it is also desirable to have the bone along a single axis and a rotation matrix to orientate it into its base position for reasons that will be discussed later. Other problems with the BVH files include the lack of calibration units, such as the scale that the joint offsets are measured in, and details about the environment, such as orientation - i.e. which direction points upwards?

5.2.2 Model System

Body motion generation models can be roughly divided into three main classes: rule-based, statistical, and learning-based. Rule based systems are based on a set of rules, as the name suggests. Statistical systems typically construct a mapping from speech to gestures based on the statistics in a given dataset of human gesticulation. Learning-based models are those whose parameters are learned from data, usually using a machine learning algorithm. In this section, we only review the learning-based models because the scope of this thesis is based on neural network model and the others two has been implicitly mentioned in Chapter 2.

At first sight, the proposed method appears to be similar to the frame-based speech-to-motion mapping with encoder-decoder DNN proposed by Kucherenko et al. [2019]. The authors applied representation learning to learn a motion embedding z with the auto-encoder, and then learnt a mapping from the speech features s to the learnt motion representation z with DNN. The synthesised motion was generated by converting the predicted z through the decoder. In this work, we built two auto-encoders for speech and motion respectively, motivated by the fact that some auto-encoder architectures exhibit good performance in information compression. We followed the same idea

that predicts motion embedding and converts the predicted motion embedding to the final motion output through the motion decoder. However, we mapped from the speech embedding. Furthermore, we employed the same idea of our previous work DCCCAE in the two auto-encoders, which train with MSE loss and CCA loss between both targets.

Previous works have proposed to use not only speech to estimate motion (Henter et al. [2020]) but also text (Yoon et al. [2019]) or the current motion (Ghosh et al. [2017]) to predict the future motion. These methods demonstrate the potential of using different types of input to predict body motion in frame-based systems, but body motion is a continuous and temporal data type. Our proposed method predicts the embedding in the frame-based system and converts the predicted motion embedding back to sequential format.

To capture the temporal information of the input and output streams in motion estimation, RNN has been proposed. Ginosar et al. [2019] reported the results of generating motion sequence in a GAN-RNN system. The proposed generative model learnt to predict the temporal stack of poses from the given audio input, while an adversarial discriminator ensured that the predicted motion was both temporally coherent and in the style of the speaker. However, Hernandez et al. [2019] pointed out that the RNN-based methods often suffer from error accumulation and thus are not good at predicting long-term human motion, where a conversation scenario usually last for more than five minutes. Thus, our proposed method did not consider RNN-based methods in both auto-encoders and used FNN-based methods only.

Li et al. [2021] proposed latent code learning to resolve the one-to-many mapping between audio and body motions. They used random sampling to generate the latent code to replace the motion-specific feature and concatenated with shared features extracted from speech to estimate body motion. In our work, we did not consider this one-to-

many mapping problem because we assumed that in real-life conversation, there are definitely differences in the speech signal in terms of voice, emotion and so on with the same utterances. This one-to-many mapping problem only occurs in laboratory experiments as the input is always the same, and it is assumed that there are many possible outputs.

5.3 Dataset Description

5.3.1 Trinity

We were provided with the Trinity Speech-Gesture Dataset (Ferstl and McDonnell [2018]) as the database for the GENEA2020 challenge. A male native English speaker was involved in the collection of the dataset. For the audio, the actor produced spontaneous and natural conversational speech without interruptions, that is, without verbal cues from a conversation partner. Moreover, the actor chose the topic he would like to speak on in the conversation with a happy disposition and included a large quantity of gesture motions. Each recording was approximately 10 minutes long. The author captured 23 takes, totalling 244 minutes of data (provided for training in the challenge). The author captured the actor's motion with a 53 marker setup and 20 Vicon cameras at 59.95 frames per second (FPS). The audio was recorded at 44 kHz.

Speech Feature: First, we down-sampled the audio rate from 44 kHz to 4 kHz. Raw wave-form vectors were extracted with a window of 125 ms and 67 ms shifting, which resulted in 500 dimensions. The reason for using such an unusual time window is to have the same number of data points as the following OpenSMILE MFCC extraction. Furthermore, we extracted the MFCC12_E_D_A feature set from OpenSMILE toolkit. This configuration extracted MFCCs from 100 ms audio frames (sampled at a rate of 50 ms) (Hamming window). It computed 12 MFCCs (1-12) from 26 mel-frequency bands and applied a cepstral liftering filter with a weight parameter of 22, and the log-

energy was appended. The 13 delta and 13 acceleration coefficients were appended to the features as well.

Body Motion: The motion data was stored in the BioVision Hierarchy (BVH) format. The BVH data describes motion as a time sequence of Euler rotations for each joint in the defined skeleton hierarchy. In the present study, these Euler angles were converted to a total of 69 global joint positions in 3D, shown in Figure 5.4. We extracted the upper body motion only, which included 15 out of 69 global joint positions and excluded the finger joints. This refers to the joints shown in Figure 5.4, which are above the '0'. Since each joint is under Euler angles representation, this means that the dimension of our body motion is 45. Some recordings had a different frame rate than others; therefore, we down-sampled all recordings to a common frame rate of 20 FPS as well as matching the frame rate of the audio. For the purpose of fast convergence in training, we applied standard normalisation (zero mean and unit variant) to the data at each rotation of the joints.

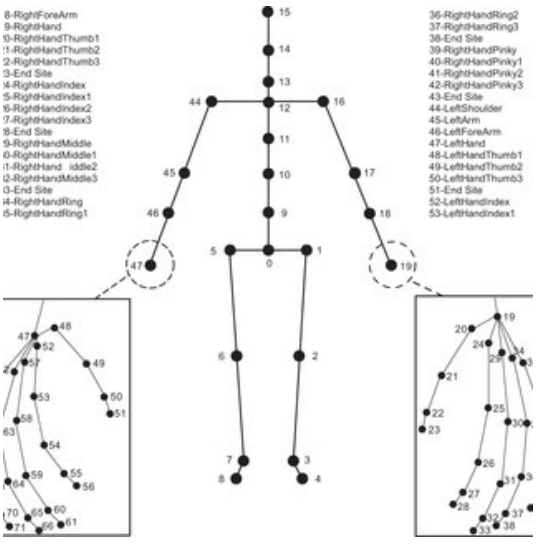


Figure 5.4: A sample of 69 joints in BVH format

5.4 Methodology

In this section, we will first describe the definition of the frame-based system, carrying on with the pros and cons of the system. After that, we propose a sequential system in a frame-based manner. We describe and discuss how the proposed system resolves the problems in the standard frame-based system. Finally, the experimental setup is detailed as well.

5.4.1 Frame-Based System

A frame-based system refers to a model thattakes a data point, which consists of many different features, and then outputs only one corresponding result. Accordingly, in our thesis, the model takes a sliding window of speech information and outputs only one frame of motion data. The predicted motion data is then concatenated together, forming a video of motion movement. Examples of this include the work we had done in Chapter 4 and the work from other researchers (Henter et al. [2020], Yoon et al. [2019], Ghosh et al. [2017]). This frame-based system can be found in other speech-related tasks as well (Hinton et al. [2012], Dahl et al. [2012], Yu et al. [2012], Jaitly et al. [2012]).

There are pros and cons in the frame-based system. First of all, the frame-based system is easy to construct. The data processing procedure is fast, and the architecture of the system is simple. Producing a prototype and analyzing the results is done rapidly. Second, the hardware requirement of a frame-based system for training and inference is much lesser than a sequential system. Thus, students and researchers can easily use their personal computer/free cloud platforms to assess the performance. However, we can not ignore the cons of the frame-based system. In speech-related tasks, both input and output data are sequential. With a frame-by-frame prediction, the system does not consider the temporal information of the data without any recurrent unit. Of course,

we can modify the system to take multiple frames of input and then output multiple predictions. In such a case, the model can learn the temporal information, but the complexity of the system increases dramatically.

5.4.2 Proposed System

To resolve the cons and maintain the pros of the standard frame-based system, we proposed an embedded prediction procedure in our system. Our proposed system can be separated into three modules: 1) D-DCCCAE for compressing the high-dimension input (e.g., waveform, body motion) to the distributed embedding of low dimensions, 2) a regression model for predicting the sequential motion embedding from the wave embedding and 3) a post-filtering auto-encoder for reconstructing smooth head motion. The overall framework of our proposed model is shown in Figure 5.5. Since the post-filtering is the same as what we described in Chapter 3 and Chapter 4, we will not describe it in the following section again.

5.4.2.1 Double DCCCAE

In our previous work, we compressed high-dimension waveforms to low-dimension and correlated embedding with head motion using a single auto-encoder of CorrNN(Lu and Shimodaira [2020]). However, our work here is different from the aforementioned research studies, in which Chandar et al. [2016], Wang et al. [2015] compressed the two streams into one common and correlated space using two auto-encoders; on the other hand, we proposed compressing the streams into different spaces with different correlated objects. We expanded our work here to apply two CorrNN auto-encoders since the dimension of the body motion in this work is much higher than the head motion in our previous work. We compressed the information into fixed-length embeddings. Thus, we employed two auto-encoders in which the hidden layers were trained in such a way as to not only minimise the reconstruction error but also maximise the canoni-

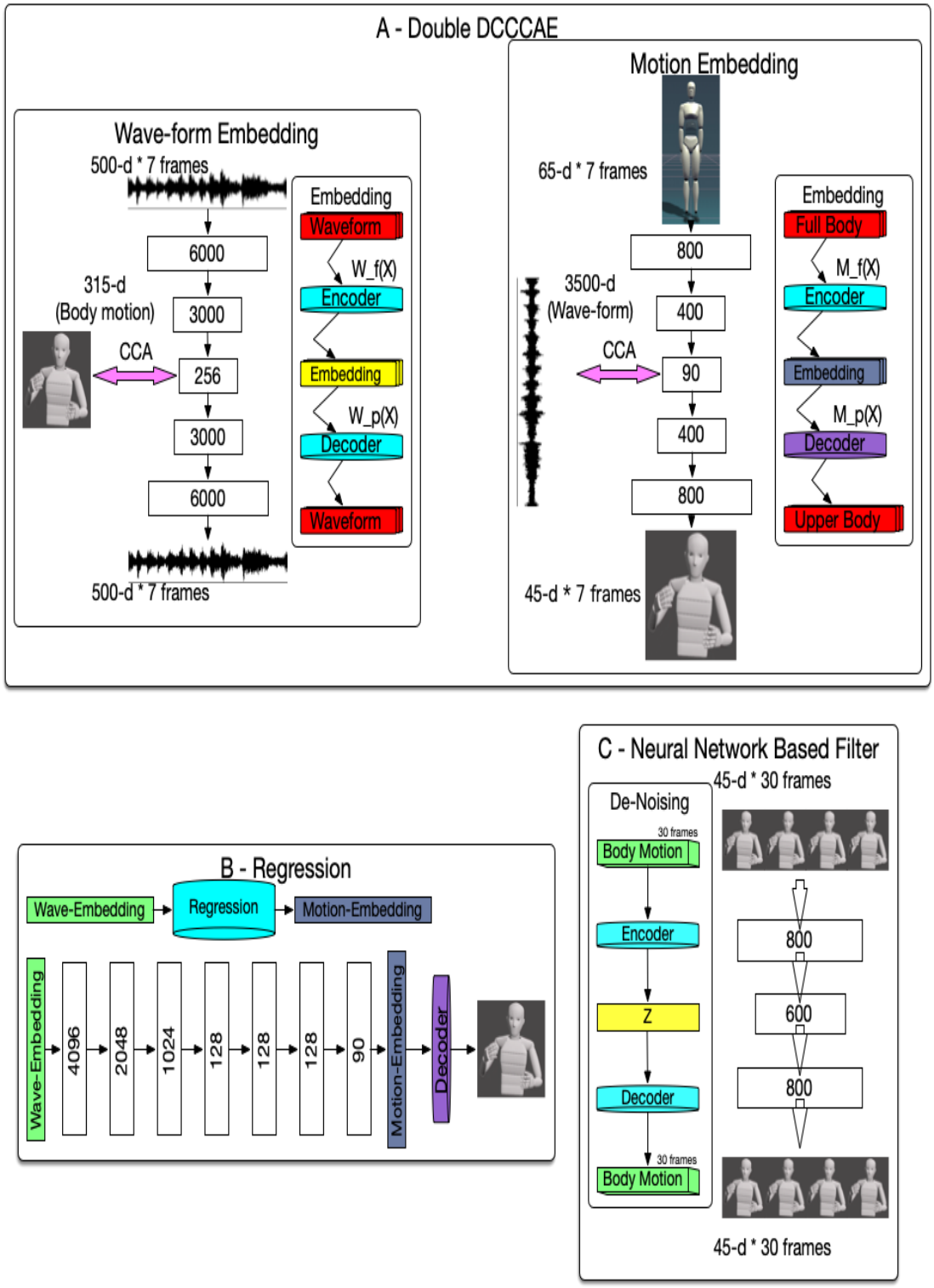


Figure 5.5: Overview of the proposed system comprised of three modules: (A) embedding with double-DCCCAE, (B) DNN-based sequential motion embedding regression from the waveform embedded features, (C) post-filter with an autoencoder.

cal correlation with body motion, shown in Figure 5.5(A). Thus, instead of projecting the two features to a common subspace, we projected the two features to two separate subspaces to ensure that the embedded features were well correlated with the objective features.

We trained each proposed DCCCAE with the following objective function:

$$\begin{aligned} \text{Obj}_{\text{DCCCAE}} = \sum_{t} \|\boldsymbol{X}_{t\pm3} - p(f(\boldsymbol{X}_{t\pm3}))\|^2 \\ -\text{CCA}\left(f(\boldsymbol{X}_{t\pm3}), \boldsymbol{Y}_{t\pm3}\right) \end{aligned} \tag{5.6}$$

In the above equation, $\boldsymbol{X}_{t\pm3}$ represents the input feature vector at a time instance t to the encoder, $f(\,)$ represents the projection with the encoder, $p(\,)$ represents the reconstruction with the decoder and $\boldsymbol{X}$ and $\boldsymbol{Y}$ denote the whole sequences of feature vectors and objective feature vectors, respectively.

5.4.2.2 Regression Model

The idea of predicting motion embedding from speech was proposed by Kucherenko et al. [2019]. This framework first applies representation learning to learn a motion representation in a frame-based system. Furthermore, it encodes speech to the learnt motion representation and decodes the same through the motion decoder. We expanded this idea to a frame-based model in a sequential manner with our highly correlated features estimated by the proposed D-DCCCAE. We mapped a frame of waveform embedding to a frame of motion embedding and decoded through the motion decoder, shown in Figure 5.5(B). The decoded motion was in a sequence of multiple frames.

As shown in Figure 5.5(B), the wave-embedding is 256 dimensions and motion-embedding is 90 dimensions. Both of them are extracted from Figure 5.5(A). Last, the output of the regression is 1350 dimensions, which are 30 frames of the body motion stacking together.

A simple feed-forward deep neural network was applied here for the regression from the wave-form embedded vector to the motion-embedded vector. We did not consider RNN (e.g., LSTM, GRU) because the present study focuses on decoding a sequential motion movement from a frame-based embedding vector, and the framed-based mapping between the two embedded features does not have a temporal relationship. Another reason is that the calculation of inputting a single frame of compressed feature into RNN is the same as inputting the data into a FNN layer, thus there is no reason to build an RNN system.

5.4.2.3 Post-Filtering

Under our hypothesis, the decoded motion should not require any post-filtering process as if the motion decoder is trained well. Even though we had trained the motion decoder with a tiny reconstruction error, the decoded motion in Figure 5.5(B) is still noisy. There are possible reasons, (1) the trade-off between the reconstruction error objective and the CCA objective during training. This causes the decoder hardly to be perfect in either way. (2) The generated trajectories have movement with minor jerkiness due to the fast frequency of the speech as mentioned in Chapter 1. (3) The two joining points between the two frames require smoothing. Thus, we trained a neural-network-based post-filter to overcome these problems in the present study Lu and Shimodaira [2020], Kucherenko et al. [2019]. This filter is similar to the one we developed for the head motion estimation in Chapter 3. The only difference is that the input and output contain more rotation joints, which are 1350 dimensions. This difference in data dimension causes the width of each layer to be different as well, shown in Figure 5.5(C).

5.4.3 Experiment Setup

As the challenge only provided a total of 23 training files, we extracted 25 seconds of the video-audio data from the middle of each provided training file, totalling about 9.5 minutes as the validation data, and the rest of the data were used in training. For the testing data, another 10 audio files (with transcripts), totalling about 20 minutes, were provided from the challenge without the motion data. Thus, we could not do any objective evaluation of the test data because we did not have the ground truth of the motion data for the test data.

We conducted preliminary experiments to decide the depth and width of the D-DCCCAE, regression models and the post-filter AE, which are shown in Figure 5.5. Training was conducted on a GPU machine and a multi-CPU machine with Pytorch version 1.5 by mini-batch training using Adam optimisation (learning rate 0.0002) (Kingma and Ba [2015]). The batch size was 4096, and the epoch was 500. Finally, the motion-decoder was fine-tuned while training with the regression model.

In the evaluation, test data was fed to the trained regression model, and motion embedding was predicted frame by frame and converted to sequential frames through the motion-decoder. Afterwards, the output of the prediction model was then combined to form distinct body motion with the overlap-add method and concatenation of 30 time frames, which were fed to the post-filtering autoencoder. The final output for animation was generated with the overlap-add method again.

5.5 Results

In this section, there are local and remote evaluations. The local evaluation refers to that we make a comparison with the stacking level of the features. Then, the regression models trained with the stacking features are evaluated with NMSE and local CCA in the validation dataset. For remote evaluation, we submitted our system output files

Table 5.2: Local CCA of stacking multiple frames between speech information and body motion. Width refers to the stacks of frames.

Feature	Width	CCA	
		Train	Valid
Wave-form	1	0.624	0.631
	3	0.483	0.490
	5	0.418	0.426
	7	0.	0.004
MFCC	1	0.481	0.481
	3	0.588	0.591
	5	0.602	0.609
	7	0.566	0.574
DCCCAE	1	0.835	0.887
	7	0.687	0.750
D-DCCCAE	7	0.792	0.861

based on the testset audio to GENEA2020 workshop, the organisers performed a subjective evaluation. From the subjective results, we summary between our system and other participants' systems.

5.5.1 Feature Analysis

Again, to understand the relationship between the motions and the information within waveform and MFCC features, we performed the basic correlation analysis between speech features and body motion in local CCA. This time, we also included the feature proposed in this Chapter, D-DCCCAE, and the feature we proposed in Chapter 4, DCCCAE. Table 5.2 highlights that the raw waveform feature has a weaker correlation with body motion when stacking more frames. The correlation with MFCC features remains in the range between 0.4 and 0.5. Our proposed two embedded features achieved the highest correlation, a clear and large improvement over the raw waveform and MFCC. Compared to our previous method (Lu and Shimodaira [2020]), D-DCCCAE shows improvement over DCCCAE for seven frames but is comparable for one frame.

Table 5.3: Comparison of different systems in terms of performance of body motion prediction, where MSE and local CCA are calculated between predicted body motion and ground truth. '7to7' refers to using seven frames of the features to estimate seven frames of the body motion. M_X refers to the regression model trained with feature X

Model	Stack of Frame	Train		Valid	
		MSE	CCA	MSE	CCA
M_{MFCC}	7to7	0.984	0.545	1.202	0.332
M_{DCCCAE}	7to7	0.974	0.563	1.203	0.330
$M_{D-DCCCAE}$	7to7	0.989	0.510	1.203	0.334

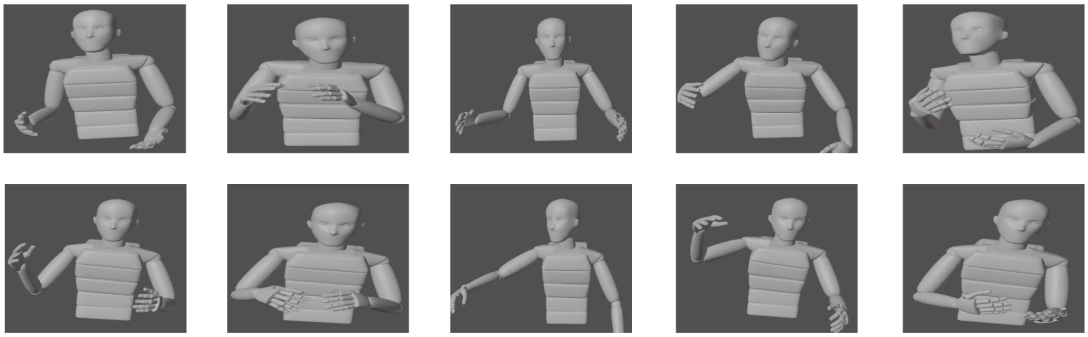
5.5.2 Motion Estimation Results

To evaluate the results objectively, we further conducted the evaluation of the body motion estimation for MFCC, DCCCAE and D-DCCCAE features. Table 5.3 shows the result of different systems with different number of the input and output frames, where were examined in MSE and local CCA. $M_{D-DCCCAE}$ has the highest MSE and lowest CCA in the train set, but achieves the highest CCA in the valid set and MSE is similar among the three models for the valid set. This indicates that our proposed model has the potential to perform better if the generalisation of the model is better.

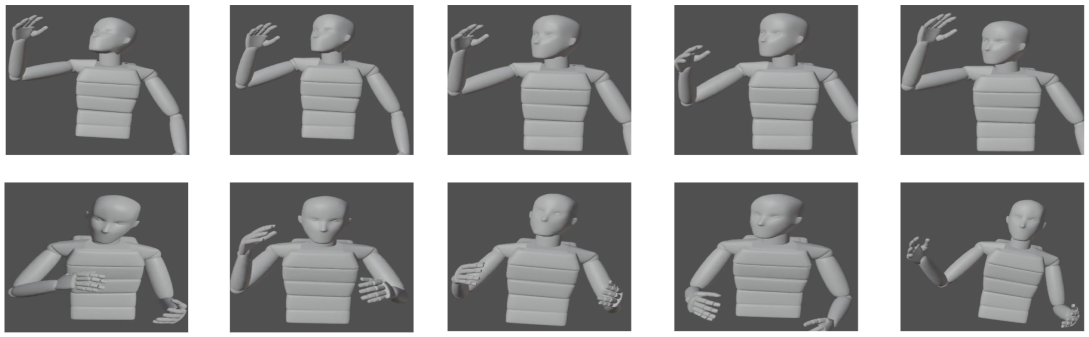
Even though we did not have the values of the motions for the test set, we received the animated videos from the GENEA2020 organisers. We produced some screenshots of the signature motions over 5 seconds from the selected videos, shown in Figure 5.6. Under the observation of the screenshots, we noticed that the proposed system might not be able to estimate the exact motions, but there are similar movement patterns over time.

5.5.3 Subjective Evaluation

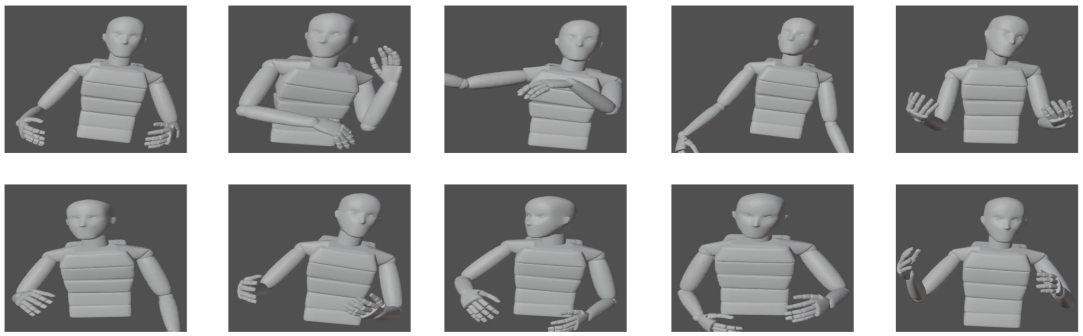
We also submitted our proposed model to the GENEA2020 challenge (Kucherenko et al. [2020]), and they conducted a perceptual test inspired by the MUSHRA test (International Telecommunication Union) through the crowd-sourcing platform Prolific (formerly Prolific Academic) in two aspects: human-likeness and appropriateness



(a) Video File 2 over 0s - 5s. An In-outward movement of hands.



(b) Video File 17 over 5s - 10s. An up-down movement of the left hand.



(c) Video File 32 over 0s - 5s. Forward and backward movements of the body and the head motions.

Figure 5.6: Screenshots of the signature motions over 5 seconds in the animated videos between the reference motions and the estimated motions. The top row is the reference video and the bottom row is the estimated one.

(Kucherenko et al. [2020]).

There were a total of nine models: five models from the participants (including us), two baseline models (Kucherenko et al. [2019], Yoon et al. [2019]), one ground truth model and one anchor model. The following abbreviations were used to represent each model in the evaluation:

- N : Ground truth.
- M : Anchor (mismatched) natural motion capture from the actor, corresponding to a different speech segment than that played together with the video. This ensures the production of very high-quality motion (same as N) but whose behaviour is completely unrelated to the speech.
- BA: The baseline system (Kucherenko et al. [2019]) that takes only speech audio into account when generating system output
- BT: The baseline system (Yoon et al. [2019]) that takes text transcript information (including word timing information) into account when generating system output
- S... : Participants' submissions (ours is SB).

The evaluation was processed such that every participant was assigned about 10 different speech segments and the corresponding generated motion videos of each segment from different systems. Furthermore, each participant was asked to watch each video and give a score on a 0- to 100-point rating scale that was divided into successive 20-point intervals, which were labelled (from best to worst) 'Excellent', 'Good', 'Fair', 'Poor', and 'Bad'. A total of 125 participants in each study were recruited and asked to follow the instructions to rate each video.

The results of the human-likeness and appropriateness evaluations are shown in Table 5.4. We are one of the two teams who used audio features only among the submis-

Table 5.4: Summary statistics of user-study ratings for all conditions in the two studies, with 0.01-level confidence intervals. The human-likeness of M was not evaluated explicitly, but is expected to be very close to N since it uses the same motion clips. F:Input feature, A: Audio feature, T: Text Feature. Our proposed system is SB.

		Human-likeness		Appropriateness	
ID	F	Median	Mean	Median	Mean
N	-	$72 \in [70, 75]$	67.6 ± 1.8	$81 \in [79, 83]$	73.8 ± 1.8
M	-	-	-	$56 \in [53, 59]$	53.3 ± 2.0
BA	A	$46 \in [44, 49]$	46.2 ± 1.7	$40 \in [38, 41]$	40.4 ± 1.8
BT	T	$55 \in [53, 58]$	54.6 ± 1.8	$38 \in [35, 40]$	38.5 ± 1.9
SA	A+T	$38 \in [35, 41]$	40.1 ± 1.9	$35 \in [31, 37]$	36.4 ± 1.9
SB ours	A	$52 \in [50, 55]$	52.8 ± 1.9	$43 \in [40, 45]$	43.3 ± 2.0
SC	A	$57 \in [55, 60]$	55.8 ± 1.9	$50 \in [48, 52]$	50.6 ± 1.9
SD	A+T	$60 \in [57, 61]$	58.8 ± 1.7	$49 \in [46, 50]$	48.1 ± 1.9
SE	A+T	$49 \in [47, 51]$	49.6 ± 1.8	$47 \in [44, 49]$	45.9 ± 1.8

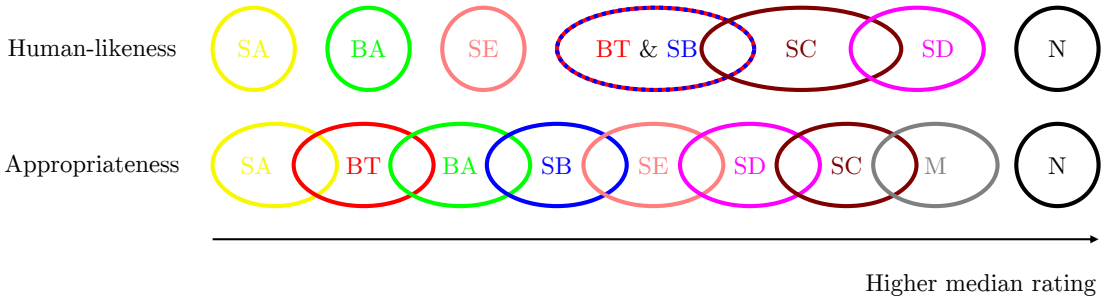


Figure 5.7: Significance of differences between conditions in the two studies. Each conditions is an ellipse; if two ellipses overlap (or, in one case, coincide), that means that the corresponding conditions were not statistically significantly different at the 0.01 level after Holm-Bonferroni correction. There is no scale on the axis here since the plot only is designed to visualise the partial ordering induced by the significance tests (i.e., ordinal information only).

sions. Our model (SB) was rated third in human-likeness and fourth in appropriateness among the participants' submissions. Interestingly, the mismatched system M ranked a higher lower bound than all synthesis systems in the appropriateness aspect. This may be explained by the high gesture rate and low amount of pauses in the dataset in combination with the known fact that the temporal alignment between speech and gesture is not exact. Moreover, the sample median score of our model was above BA but below BT in terms of the human-likeness aspect, and it was above both baselines in the appropriateness aspect. Figure 5.7 visualises the (partial) ordering of conditions induced by

the significance tests in each study. In the human-likeness aspect, our model (SB) was not statistically significantly different from BT, but better than BA. In the appropriateness aspect, there was not much difference between BA and our model, but better than BT. These results suggest that our proposed embedded features effectively improved the model generalisation compared to BA, which had a similar model structure and ideas as ours. Even though we did not make use of the text transcriptions provided with the challenge, we can observe that our system compares favourably to the BA model. A benefit that may contribute to our higher naturalness ranking than BA is our system's ability to a pair of high correlation embedded features. This pair of features enables the regression model to learn and generalise well. It is interesting to observe that our system was rated higher than SA and SE in terms of the human-likeness aspect but was rated lower than SD in terms of the appropriateness aspect, where both of the systems made the usage of audio and text information. This could be explained by the motions themselves being more smooth and more natural when without audio, this could be beneficial from the motion decoder and the post filter. However, the motions were not appropriate for the audio. Since the actor was requested to speak and react faster than usual, our model might not be able to catch the information with the current frame window.

5.6 Further Considerations

Several aspects of the work presented in this chapter warrant further discussion. First, the CCA objective function that was used for the motion embedding model in Section 5.4 aimed to maximise the correlation between the extracted motion embedding and the audio waveform. While this objective function did increase the correlation, there are some other ways to consider achieving better results in the later regression. It is possible that this learning objective was not what we wanted as we were not using waveform to predict those motion embeddings.

It may be possible to correlate both extracted embedded features instead. The benefit of correlating both extracted embedded features is that higher the correlation between both features is, the better the regression result is. A downside to correlating both embedded features is that during training, it is hard to design the architecture and the procedure to simultaneously converge both embedding models. Training both embedding models simultaneously could miss the training objective for one or both models. One of the embedded features may take over another one's convergence or both embedded features may contradict each other, causing the whole training to fail.

5.7 Conclusion

In this chapter, we expanded our previous work to propose a new architecture. The proposed model not only creates a highly correlated feature pair, but also generates sequential raw motion data in a frame-based manner. From the objective evaluation, we concluded that D-DCCCAE enables the creation of a more correlated feature pair, diminishing the side-effect of stacking multiple blocks of speech information and motion data. D-DCCCAE achieved the highest CCA in the valid set in the model comparison. In the subjective evaluation, our model achieved a higher score than the BA in both aspects (human-likeness and appropriateness) according to the participants' preferences, suggesting that the highly correlated feature pair and the sequential estimation helped improve the model generalisation. In the future, we can consider exploring higher stacking to unearth the potential of D-DCCCAE.

Chapter 6

Discussion and Conclusion

6.1 Overall Achievements

This thesis presented experiments that explore how to estimate motion from waveform and evaluate the predicted motion in a variety of measures ranging from numerical differences and distribution and subjective tests. This thesis aimed to investigate the usage of waveform to estimate motion in an end-to-end system, while this technique has been popular in ASR at the time that this thesis was conceived and written. To continue using waveform directly for ASR, it may be straightforward to extract the most useful information, where the waveform itself contains the full information of the speech. To apply the same on speech-driven motion, the system is supposed to extract the information correlated with the motion only and then estimate the corresponding motions. Each of the chapters in this thesis has addressed a different angle of the extracted features and speech-driven motion system. Some of the original motivation for the feature extraction in this work was inspired by early work in the field of representation learning, which the model learns a representation of multiple views. Unlike representation learning for multiple views, which projects to the subspaces, the objective in this thesis was to extract and correlate the useful information of the speech

waveform with the motions.

Chapter 3: Speech-driven Head Motion System with Waveform. A new approach was proposed and tested for speech-driven head motion from waveform using DCC-CAE. The presented approach was also defined as a type of representation learning because it retained the useful information, which correlates highly with head motion, through the CCA objective function. The correlated representations were created and evaluated at several levels of feature analysis and regression results in different aspects. We also proposed a new objective measure to evaluate the predicted head motion through a detection method in the frequency domain called TWV. In addition to applying common metrics (such as NMSE and local CCA), we also used movement distribution (such as velocity, acceleration and jerkness) in the final results evaluation. The evaluation results found that the proposed feature, $\text{Wav}_{\text{CCCAE}}$, is more strongly correlated with motion than Wav_{AE} and other popular spectral features such as MFCC and Fbank among different subjects; in this regard, the analysis of the features distribution among the subjects demonstrated a clear distinct cluster for each subject in the proposed feature only. In the regression results, the M_{CCCAE} achieved a better score in NMSE but worse in local CCA than M_{MFCC}; additionally, the analysis based on TWV indicated that M_{MFCC} and M_{CCCAE} were comparable performance, and the movement distribution graph indicates that M_{CCCAE} tends to produce smoother and slower movement than M_{MFCC}.

Chapter 4: LSTM-Based Head Motion Estimation with DCCCAE. An advanced architecture with several variations was introduced in Chapter 4 based on the speech-driven head motion paradigm and operating on the same data. The objective of using and modifying the head motion regression model was to improve the handling of sequential data information using LSTM-based models. Two LSTM-based models were developed based on the original FNN-based model. The two model variants were compared to the original FNN-based model for head motion synthesis. The LSTM-based

regression models were able to boost the overall performance in NMSE and CCA and adapted better with the proposed feature ($\text{Wav}_{\text{CCCAE}}$) than MFCC. Subjective tests for motion naturalness, appropriateness and model comparison as well as objective measures (NMSE and local CCA) demonstrated that it is possible to estimate head motion using speech waveform directly with the help of DCCCAE, that achieving SOTA results.

Chapter 5: Upper Body Motion Estimation Using Double-DCCCAE. In addition to generating head motion poses, we extended our framework to generate upper body motions in Chapter 5. Unlike in Chapter 4, not only did we modify the regression model but we also added another DCCCAE for the motion embedding. The objective of adding another DCCCAE and modifying the regression model was to adapt the changes of the model's output from three dimensions (X,Y,Z-trajectory) to 135 dimensions (45 joints, each joint consisting of X,Y,Z-trajectory) and encourage temporal sequence estimation in the frame-based system. Another DCCCAE for motion embedding was developed to compress motion data into lower embedded features and correlate the embedded features highly with speech waveform, motion estimation was replaced with motion embedding estimation, and then the estimated motion embedding was decoded back to sequential motion data. This D-DCCCAE was then compared to the original DCCCAE in Chapter 3 for motion objective evaluation, and we submitted our D-DCCCAE model to the GENEA2020 challenge with four other teams for subjective evaluation. The proposed D-DCCCAE achieved the highest CCA in the valid set in the model comparison. In the subjective evaluation, our model achieved a score that is higher than one of the baselines in both aspects (human-likeness and appropriateness) according to the participants' preferences.

6.2 Limitations

While the work in this thesis provides new insights into speech-driven motion and exciting avenues of future work, it is not without some limitations. Advancements in the field of encoder-decoder architectures arose after the experiments in Chapter 3 had already started based on the DCCCAE architecture. In particular, a new type of auto-encoder has since been published (Défossez et al. [2022]), as well as new generative models, such as flow (Henter et al. [2020]) and diffusion (Ho et al. [2020]). These advances were developed at famous laboratories with a multitude of other engineering resources available. Generally, interest in motion synthesis has been increasing over the past few years. Even with these recent advancements, there is no single definition or formalism that defines motion synthesis. The work presented in this thesis demonstrates why it is so difficult to estimate motion from speech. The DCCCAE method for representation learning from Chapter 3 may potentially struggle between reconstruction error and correlation (which would hardly to find a optimal point) or there may be some aspects to motion synthesis that are fundamentally limiting such as an effective objective evaluation and lack of large dataset or the one-to-many problem.

6.3 Future Work

In this section, we discuss potential directions in which this thesis could be extended. Given that the speech-driven motion system and representation learning were successful for the usage of speech waveform directly as input to estimate upper body motion, it would be interesting to explore an approach that learns correlated embedded features for facial expression synthesis as well. This research direction would just require replacing the current motion dataset with the facial expression dataset. It would be beneficial to explore a possible solution to create a highly correlated feature for all types of motions.

6.4 Concluding Remarks

The view of representation learning presented in this thesis relies on the ability to evaluate learned representations either intrinsically or as they can be applied in motion synthesis tasks. To tackle speech-driven motion, this thesis presented a feasible approach to synthesise motion from speech waveform directly by applying CCA constraints in the embedding generation. The principles of the speech-driven motion that were explored in this thesis can be used as a guide for further work on facial expression synthesis, especially for cases where speech waveform may be obtained from the application technology. We also introduced a new objective measure to evaluate the quality of the generated head motions: peak detection. We gave recommendations as to how to conduct a subjective test. Overall, by covering these areas, we have contributed to the field in a structured and experimentally supported manner. In doing so, not only did we meet the original objective of finding a new method for mapping speech to head motion but we also improved the supporting areas. Finally, speech-driven motion synthesis is an inherently difficult task, but experiments in this work demonstrated that the proposed approach enables the creation of a feature that correlates highly with motions for downstream tasks.

Bibliography

MOTU. http://www.motu.com/.

Naturalpoint Optitrack. http://www.naturalpoint.com/optitrack.

Mark I. Alpert and Robert A. Peterson. On the interpretation of canonical analysis. *Journal of Marketing Research*, 9(2):187–192, 1972. ISSN 00222437. URL http://www.jstor.org/stable/3149953.

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, Jie Chen, Jingdong Chen, Zhijie Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Ke Ding, Niandong Du, Erich Elsen, Jesse Engel, Weiwei Fang, Linxi Fan, Christopher Fougner, Liang Gao, Caixia Gong, Awni Hannun, Tony Han, Lappi Johannes, Bing Jiang, Cai Ju, Billy Jun, Patrick LeGresley, Libby Lin, Junjie Liu, Yang Liu, Weigao Li, Xiangang Li, Dongpeng Ma, Sharan Narang, Andrew Ng, Sherjil Ozair, Yiping Peng, Ryan Prenger, Sheng Qian, Zongfeng Quan, Jonathan Raiman, Vinay Rao, Sanjeev Satheesh, David Seetapun, Shubho Sengupta, Kavya Srinet, Anuroop Sriram, Haiyuan Tang, Liliang Tang, Chong Wang, Jidong Wang, Kaifu Wang, Yi Wang, Zhijian Wang, Zhiqian Wang, Shuang Wu, Likai Wei, Bo Xiao, Wen Xie, Yan Xie, Dani Yogatama, Bin Yuan, Jun Zhan, and Zhenyao Zhu. Deep speech 2 : End-to-end speech recognition in english and mandarin. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 173–182, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/amodei16.html.

M. Anderson. Nonverbal communication. In Keith Brown, editor, *Encyclopedia of Language Linguistics (Second Edition)*, pages 690–692. Elsevier, Oxford, second edition edition, 2006. ISBN 978-0-08-044854-1. doi: https://doi.org/10.1016/B0-08-044854-2/01432-2. URL https://www.sciencedirect.com/science/article/pii/B0080448542014322.

T.W. Anderson. *AN INTRODUCTION TO MULTIVARIATE STATISTICAL ANALYSIS, 3RD ED*. Wiley India Pvt. Limited, 2009. ISBN 9788126524488. URL https://books.google.co.kr/books?id=1iF0CgAAQBAJ.

Elisabeth André, Elisabetta Bevacqua, Dirk K.J. Heylen, Radoslaw Niewiadomski, Catherine Pelachaud, Christopher Peters, Isabella Poggi, and Matthias Rehm. *Non-verbal Persuasion and Communication in an Affective Agent*, pages 585–

608. Cognitive Technologies. Springer, 2011. ISBN 978-3-642-15183-5. doi: 10.1007/978-3-642-15184-2_30. 10.1007/978-3-642-15184-2$_3$0.

Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep Canonical Correlation Analysis. In Sanjoy Dasgupta and David McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 1247–1255, Atlanta, Georgia, USA, 17–19 June 2013. PMLR. URL http://proceedings.mlr.press/v28/andrew13.html.

Ayşe Arslan and Oktay Yildiz. Automated auscultative diagnosis system for evaluation of phonocardiogram signals associated with heart murmur diseases. *Gazi University Journal of Science*, 31, 03 2018.

Silèye O. Ba and Jean-Marc Odobez. A video database for head pose tracking evaluation. Idiap-Com Idiap-Com-04-2005, IDIAP, Martigny, Switzerland, 2005.

Eric Battenberg, Jitong Chen, Rewon Child, Adam Coates, Yashesh Gaur Yi Li, Hairong Liu, Sanjeev Satheesh, Anuroop Sriram, and Zhenyao Zhu. Exploring neural transducers for end-to-end speech recognition. *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 206–213, 2017.

Atef Ben Youssef, Hiroshi Shimodaira, and David Braude. Speech driven talking head from estimated articulatory features. In *Proc. ICASSP*, pages 4573–4577, 2014.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: a review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798—1828, August 2013. ISSN 0162-8828. doi: 10.1109/tpami.2013.50. URL https://doi.org/10.1109/TPAMI.2013.50.

Christian Benoît, Martine Grice, and Valérie Hazan. The sus test: A method for the assessment of text-to-speech synthesis intelligibility using semantically unpredictable sentences. *Speech Communication*, 18(4):381–392, 1996. ISSN 0167-6393. doi: https://doi.org/10.1016/0167-6393(96)00026-X. URL https://www.sciencedirect.com/science/article/pii/016763939600026X.

R.L. Birdwhistell. *Introduction to Kinesics: An Annotation System for Analysis of Body Motion and Gesture*. Department of State, Foreign Service Institute, 1952. URL https://books.google.co.uk/books?id=Ad99AAAAMAAJ.

Matthew B. Blaschko and Christoph H. Lampert. Correlational spectral clustering. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. doi: 10.1109/CVPR.2008.4587353.

Bruce P. Bogert. The quefrency analysis of time series for echoes : cepstrum, pseudo-autocovariance, cross-cepstrum and saphe cracking. 1963. URL https://api.semanticscholar.org/CorpusID:59352135.

Maarten Boksem, Theo Meijman, and Monicque Lorist. Effects of mental fatigue on attention: An erp study. *Brain research. Cognitive brain research*, 25:107–16, 10 2005. doi: 10.1016/j.cogbrainres.2005.04.011.

D. Bolinger and D.L.M. Bolinger. *Intonation and Its Parts: Melody in Spoken English*. Stanford University Press, 1986. ISBN 9780804712415. URL https://books.google.co.uk/books?id=FHuAuCEs-6UC.

Dwight Bolinger. Intonation and gesture. *American Speech*, 58(2):156–174, 1983. ISSN 00031283, 15272133. URL http://www.jstor.org/stable/455326.

C. Breazeal, C. D. Kidd, A. L. Thomaz, G. Hoffman, and M. Berlin. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 708–713, 2005.

Sabine Buchholz and Javier Latorre. Crowdsourcing preference tests, and how to detect cheating. In *INTERSPEECH*, pages 3053–3056. ISCA, 2011. URL http://dblp.uni-trier.de/db/conf/interspeech/interspeech2011.html#BuchholzL11.

Carlos Busso and Shrikanth Narayanan. Interrelation Between Speech and Facial Gestures in Emotional Utterances: A Single Subject Study. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(8):2331–2347, November 2007. ISSN 1558-7916. doi: 10.1109/TASL.2007.905145.

Carlos Busso, Zhigang Deng, Ulrich Neumann, and Shrikanth Narayanan. Natural head motion synthesis driven by acoustic prosodic features: Virtual Humans and Social agents. *Computer Animation and Virtual Worlds*, 16:283–290, July 2005.

Carlos Busso, Zhigang Deng, Michael Grimm, Ulrich Neumann, and Shrikanth Narayanan. Rigid head motion in expressive speech animation: Analysis and synthesis. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(3):1075–1086, 2007a. doi: 10.1109/TASL.2006.885910.

Carlos Busso, Ulrich Neumann, and Shrikanth Narayanan. *Learning Expressive Human-Like Head Motion Sequences from Speech*, pages 113–131. 10 2007b. ISBN 978-1-84628-906-4. doi: 10.1007/978-1-84628-907-1_6.

Ruth Campbell. Review of "perceiving talking faces: From speech perception to a behavioral principle" by dominic w. massaro. *Pragmatics amp; Cognition*, 8(1): 261–264, 2000. ISSN 0929-0907. doi: https://doi.org/10.1075/pc.8.1.12cam. URL https://www.jbe-platform.com/content/journals/10.1075/pc.8.1.12cam.

Justine Cassell, Catherine Pelachaud, Norman Badler, Mark Steedman, Brett Achorn, Tripp Becket, Brett Douville, Scott Prevost, and Matthew Stone. Animated conversation: Rule-based generation of facial expression, gesture amp; spoken intonation for multiple conversational agents. SIGGRAPH '94, page 413–420, New York, NY, USA, 1994. Association for Computing Machinery. ISBN 0897916670. doi: 10.1145/192161.192272. URL https://doi.org/10.1145/192161.192272.

William Chan, Navdeep Jaitly, Quoc V. Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *ICASSP*, 2016. URL http://williamchan.ca/papers/wchan-icassp-2016.pdf.

Sarath Chandar, Mitesh M. Khapra, Hugo Larochelle, and Balaraman Ravindran. Correlational Neural Networks. *Neural Computation*, 28(2):257–285, 02 2016. ISSN

0899-7667. doi: 10.1162/NECO_a_00801. URL https://doi.org/10.1162/NECO_a_00801.

Kamalika Chaudhuri, Sham M. Kakade, Karen Livescu, and Karthik Sridharan. Multi-view clustering via canonical correlation analysis. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 129–136, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. doi: 10.1145/1553374.1553391. URL https://doi.org/10.1145/1553374.1553391.

Kyoungho Choi, Ying Luo, and Jenq-Neng Hwang. Hidden markov model inversion for audio-to-visual conversion in an mpeg-4 facial animation system. *J. VLSI Signal Process. Syst.*, 29(1–2):51–61, aug 2001. ISSN 0922-5773.

Sharada V Chougule, Mahesh S Chavan, and M S Gaikwad. Filter bank based cepstral features for speaker recognition. In *2014 IEEE Global Conference on Wireless Computing Networking (GCWCN)*, pages 102–106, 2014. doi: 10.1109/GCWCN.2014.7030857.

Ruth Breckinridge Church, Martha W. Alibali, and Spencer D. Kelly. "chapter 6. the function of gesture in learning and memory" in why gesture?: How the hands function in speaking, thinking and communicating. 2017. URL https://api.semanticscholar.org/CorpusID:67683642.

CMU. Carnegie-mellon motion capture (mocap) database (2003). URL http://mocap.cs.cmu.edu.

Daniel Cudeiro, Timo Bolkart, Cassidy Laidlaw, Anurag Ranjan, and Michael J. Black. Capture, learning, and synthesis of 3d speaking styles. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10093–10103, 2019.

Howard D. Curtis. Chapter 9 - rigid body dynamics. In Howard D. Curtis, editor, *Orbital Mechanics for Engineering Students (Third Edition)*, pages 459–542. Butterworth-Heinemann, Boston, third edition edition, 2014. ISBN 978-0-08-097747-8. doi: https://doi.org/10.1016/B978-0-08-097747-8.00009-8. URL https://www.sciencedirect.com/science/article/pii/B9780080977478000098.

Thomas L. Curtright, David B. Fairlie, and C. Zachos. A compact formula for rotations as spin matrix polynomials. *Symmetry Integrability and Geometry-methods and Applications*, 10:084, 2014.

George E. Dahl, Dong Yu, Li Deng, and Alex Acero. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):30–42, 2012. doi: 10.1109/TASL.2011.2134090.

PAUL B. DAVENPORT. Rotations about nonorthogonal axes. *AIAA Journal*, 11(6):853–857, 1973. doi: 10.2514/3.6842. URL https://doi.org/10.2514/3.6842.

Jan Peter de Ruiter. *The production of gesture and speech*, page 284–311. Language Culture and Cognition. Cambridge University Press, 2000. doi: 10.1017/CBO9780511620850.018.

D. DeCarlo, C. Revilla, M. Stone, and J.J. Venditti. Making discourse visible: coding and animating conversational facial displays. In *Proceedings of Computer Animation 2002 (CA 2002)*, pages 11–16, 2002. doi: 10.1109/CA.2002.1017501.

James Diebel. Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix*, 58(15-16):1–35, 2006.

Chuang Ding, Lei Xie, and Pengcheng Zhu. Head motion synthesis from speech using deep neural networks. *Multimedia Tools and Applications*, 74(22):9871–9888, November 2015a. ISSN 1573-7721. doi: 10.1007/s11042-014-2156-2. URL https://doi.org/10.1007/s11042-014-2156-2.

Chuang Ding, Pengcheng Zhu, and Lei Xie. BLSTM neural networks for speech driven head motion synthesis. In *INTERSPEECH*, 2015b.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022.

Florian Eyben, Martin Wöllmer, and Björn Schuller. Opensmile: The Munich Versatile and Fast Open-source Audio Feature Extractor. In *Proceedings of the 18th ACM International Conference on Multimedia*, pages 1459–1462, 2010. ISBN 978-1-60558-933-6. doi: 10.1145/1873951.1874246. URL http://doi.acm.org/10.1145/1873951.1874246.

Mireille Fares, Catherine Pelachaud, and Nicolas Obin. Transformer Network for Semantically-Aware and Speech-Driven Upper-Face Generation. In *EUSIPCO*, Belgrade, Serbia, August 2022. URL https://hal.archives-ouvertes.fr/hal-03677459.

Gaëlle Ferré, Roxane Bertrand, and Philippe Blache. The CID Video Corpus: A Multimodal Resource for Gesture Studies. In *Third ISGS conference 'Integrating Gestures'*, Evanston, Illinois, United States, 2007. URL https://hal.science/hal-00666074.

Ylva Ferstl 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 '18, page 93–98, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450360135. doi: 10.1145/3267851.3267898. URL https://doi.org/10.1145/3267851.3267898.

Jonathan Fiscus, Jerome Ajot, John Garofolo, and George Doddington. Results of the 2006 spoken term detection evaluation. ACM SIGIR Conference, Amsterdam, NL, 1970.

L. Flórez-Valencia and M. Orkisz. Chapter 6 - right generalized cylinder model for vascular segmentation. In Simone Balocco, Maria A. Zuluaga, Guillaume Zahnd, Su-Lin Lee, and Stefanie Demirci, editors, *Computing and Visualization for Intravascular Imaging and Computer-Assisted Stenting*, The Elsevier and MICCAI Society Book Series, pages 131–156. Academic Press, 2017. ISBN 978-0-12-811018-8. doi: https://doi.org/10.1016/B978-0-12-811018-8.00006-0. URL https://www.sciencedirect.com/science/article/pii/B9780128110188000060.

Felix Alexander Gers. Long short-term memory in recurrent neural networks. 2001. URL https://api.semanticscholar.org/CorpusID:144707025.

Pegah Ghahremani, Vimal Manohar, Daniel Povey, and Sanjeev Khudanpur. Acoustic Modelling from the Signal Domain Using CNNs. In *INTERSPEECH*, pages 3434–3438, 2016.

P. Ghosh, J. Song, E. Aksan, and O. Hilliges. Learning human motion models for long-term predictions. In *2017 International Conference on 3D Vision (3DV)*, pages 458–466, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society. doi: 10.1109/3DV.2017.00059. URL https://doi.ieeecomputersociety.org/10.1109/3DV.2017.00059.

S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, and J. Malik. Learning individual styles of conversational gesture. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2019.

Susan Goldin-Meadow and Martha Wagner Alibali. Gesture's role in speaking, learning, and creating language. *Annual Review of Psychology*, 64(1):257–283, 2013. doi: 10.1146/annurev-psych-113011-143802. URL https://doi.org/10.1146/annurev-psych-113011-143802. PMID: 22830562.

Susan Goldin-Meadow, Howard C. Nusbaum, Spencer D. Kelly, and Susan M. Wagner. Explaining math: Gesturing lightens the load. *Psychological Science*, 12:516 – 522, 2001. URL https://api.semanticscholar.org/CorpusID:538814.

H.P. Graf, E. Cosatto, V. Strom, and Fu Jie Huang. Visual prosody: facial movements accompanying speech. In *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition*, pages 396–401, 2002. doi: 10.1109/AFGR.2002.1004186.

Jean Ann Graham and Michael Argyle. A cross-cultural study of the communication of extra-verbal meaning by gesture. *International Journal of Psychology*, 10:57–67, 1975.

Jonathan Gratch, Anna Okhmatovskaia, Francois Lamothe, Stacy Marsella, Mathieu Morales, R. J. van der Werf, and Louis-Philippe Morency. Virtual rapport. In Jonathan Gratch, Michael Young, Ruth Aylett, Daniel Ballin, and Patrick Olivier, editors, *Intelligent Virtual Agents*, pages 14–27, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-37594-4.

Alex Graves. Generating sequences with recurrent neural networks. *ArXiv*, abs/1308.0850, 2013. URL https://api.semanticscholar.org/CorpusID:1697424.

Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, page II–1764–II–1772. JMLR.org, 2014.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent

neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 369–376, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933832. doi: 10.1145/1143844.1143891. URL https://doi.org/10.1145/1143844.1143891.

David Greenwood, Stephen Laycock, and Iain Matthews. Predicting head pose from speech with a conditional variational autoencoder. pages 3991–3995, 08 2017. doi: 10.21437/Interspeech.2017-894.

MICHAEL GRESHKO. Meet sophia, the robot that looks almost human. *National Geographic*, 2018. URL https://www.nationalgeographic.com/photography/article/sophia-robot-artificial-intelligence-science.

Kathrin Haag and Hiroshi Shimodaira. The University of Edinburgh Speaker Personality and MoCap Dataset. In *Facial Analysis and Animation Proceedings, Vienna*, 2015.

Kathrin Haag and Hiroshi Shimodaira. Bidirectional LSTM Networks Employing Stacked Bottleneck Features for Expressive Speech-Driven Head Motion Synthesis. In *Intelligent Virtual Agents*, pages 198 – 207, 2016. ISBN 978-3-319-47665-0.

U. Hadar, T.J. Steiner, E.C. Grant, and F.C. Rose. Head movement correlates of juncture and stress at sentence level. *Language and Speech*, 26(2):117–129, 1983.

J.A. Hall. Nonverbal communication, social psychology of. In Neil J. Smelser and Paul B. Baltes, editors, *International Encyclopedia of the Social Behavioral Sciences*, pages 10702–10706. Pergamon, Oxford, 2001. ISBN 978-0-08-043076-8. doi: https://doi.org/10.1016/B0-08-043076-7/01813-1. URL https://www.sciencedirect.com/science/article/pii/B0080430767018131.

David Hardoon, Janaina Mourão-Miranda, Michael Brammer, and John Shawe-Taylor. Unsupervised analysis of fmri data using kernel canonical correlation. *NeuroImage*, 37:1250–9, 11 2007. doi: 10.1016/j.neuroimage.2007.06.017.

Taniya Hasija, Virender Kadyan, Kalpna Guleria, Abdullah Alharbi, Hashem Alyami, and Nitin Goyal. Prosodic feature-based discriminatively trained low resource speech recognition system. *Sustainability*, 14(2), 2022. ISSN 2071-1050. doi: 10.3390/su14020614. URL https://www.mdpi.com/2071-1050/14/2/614.

Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Trans. Graph.*, 39(6), nov 2020. ISSN 0730-0301. doi: 10.1145/3414685.3417836. URL https://doi.org/10.1145/3414685.3417836.

A. Hernandez, J. Gall, and F. Moreno. Human motion prediction via spatio-temporal inpainting. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7133–7142, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society. doi: 10.1109/ICCV.2019.00723. URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00723.

Dirk Heylen. Challenges ahead: Head movements and other social acts in conversations. In Lynn Halle, Peter Wallis, Sarah Woods, Stacy Marsella, Catherine Pelachaud, and Dirk K.J. Heylen, editors, *Proceedings of the Joint Symposium on Virtual Social*

Agents, pages 45–52. The Society for the Study of AI and the Simulation of Behav., 2005. ISBN 1-902956-49-2. Joint Symposium on Virtual Social Agents, AISB 2005 Convention : Social Intelligence and Interaction in Animals, Robots and Agents, AISB ; Conference date: 12-04-2005 Through 15-04-2005.

Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012. doi: 10.1109/MSP.2012.2205597.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

S. Hochreiter. Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut für Informatik, Lehrstuhl Prof. Brauer, Technische Universität München, 1991.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9 (8):1735–1780, November 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Gregor Hofer and Hiroshi Shimodaira. Automatic head motion prediction from speech data. pages 722–725, 08 2007. doi: 10.21437/Interspeech.2007-299.

Gregor Hofer, Hiroshi Shimodaira, and Junichi Yamagishi. Speech driven head motion synthesis based on a trajectory model. In *ACM SIGGRAPH 2007 Posters*, SIGGRAPH '07, page 86–es, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781450318280. doi: 10.1145/1280720.1280814. URL https://doi.org/10.1145/1280720.1280814.

Gregor Otto Hofer. Speech-driven animation using multi-modal hidden markov models. 2010. URL https://era.ed.ac.uk/handle/1842/3786.

Autumn B. Hostetter and Martha W. Alibali. Visible embodiment: Gestures as simulated action. *Psychonomic Bulletin & Review*, 15:495–514, 2008. URL https://api.semanticscholar.org/CorpusID:371637.

Autumn B. Hostetter and Martha W. Alibali. Gesture as simulated action: Revisiting the framework. *Psychonomic Bulletin & Review*, 26:721–752, 2018. URL https://api.semanticscholar.org/CorpusID:54587999.

Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936. ISSN 00063444. URL http://www.jstor.org/stable/2333955.

Wei-Ning Hsu, Yu Zhang, and James Glass. Unsupervised learning of disentangled and interpretable representations from sequential data. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 1876–1887, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Lixing Huang, Louis-Philippe Morency, and Jonathan Gratch. Virtual rapport 2.0. In Hannes Högni Vilhjálmsson, Stefan Kopp, Stacy Marsella, and Kristinn R. Thórisson, editors, *Intelligent Virtual Agents*, pages 68–79, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-23974-8.

Radiocommunication Sector International Telecommunication Union. Recommendation itu-r bs.1534: Method for the subjective assessment of intermediate quality level of coding systems. URL https://www.itu.int/dms_pubrec/itu-r/rec/bs/R-REC-BS.1534-1-200301-S!!PDF-E.pdf.

C. T. Ishi, J. Haas, F. P. Wilbers, H. Ishiguro, and N. Hagita. Analysis of head motions and speech, and head motion control in an android. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 548–553, Oct 2007. doi: 10.1109/IROS.2007.4399335.

Navdeep Jaitly, Patrick Nguyen, Andrew Senior, and Vincent Vanhoucke. Application of pretrained deep neural networks to large vocabulary speech recognition. In *Proceedings of Interspeech 2012*, 2012.

Ye Jia, Yu Zhang, Ron J. Weiss, Quan Wang, Jonathan Shen, Fei Ren, Zhifeng Chen, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, and Yonghui Wu. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 4485–4495, Red Hook, NY, USA, 2018. Curran Associates Inc.

Patrik Jonell, Taras Kucherenko, Ilaria Torre, and Jonas Beskow. Can we trust online crowdworkers? comparing online and offline participants in a preference test of virtual agents. In *Proceedings of the 20th ACM International Conference on Intelligent Virtual Agents*, IVA '20, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450375863. doi: 10.1145/3383652.3423860. URL https://doi.org/10.1145/3383652.3423860.

J.A. Jones. Course 10 - nuclear magnetic resonance quantum computation. In Daniel Estève, Jean-Michel Raimond, and Jean Dalibard, editors, *Quantum Entanglement and Information Processing*, volume 79 of *Les Houches*, pages 357–400. Elsevier, 2004. doi: https://doi.org/10.1016/S0924-8099(03)80034-3. URL https://www.sciencedirect.com/science/article/pii/S0924809903800343.

Sham M. Kakade and Dean P. Foster. Multi-view regression via canonical correlation analysis. In Nader H. Bshouty and Claudio Gentile, editors, *Learning Theory*, pages 82–96, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. ISBN 978-3-540-72927-3.

Hideki Kawahara. Straight, exploitation of the other aspect of vocoder: Perceptually isomorphic decomposition of speech sounds. *Acoust Sci Technol*, 27349, 11 2006. doi: 10.1250/ast.27.349.

Adam Kendon. Gesticulation and speech: Two aspects of the process of utterance. 1981. URL https://api.semanticscholar.org/CorpusID:59859414.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In

Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Sotaro Kita. Cross-cultural variation of speech-accompanying gesture: A review. *Language and Cognitive Processes*, 24(2):145–167, 2009. doi: 10.1080/01690960802586188. URL https://doi.org/10.1080/01690960802586188.

Sotaro Kita, Martha W. Alibali, and Mingyuan Chu. How do gestures influence thinking and speaking? the gesture-for-conceptualization hypothesis. *Psychological Review*, 124:245–266, 2017. URL https://api.semanticscholar.org/CorpusID:23370881.

Sotaro Kita and Asli Özyürek. What does cross-linguistic variation in semantic coordination of speech and gesture reveal?: Evidence for an interface representation of spatial thinking and speaking. *Journal of Memory and Language*, 48(1):16 – 32, 2003. ISSN 0749-596X. doi: https://doi.org/10.1016/S0749-596X(02)00505-3. URL http://www.sciencedirect.com/science/article/pii/S0749596X02005053.

Aniket Kittur, Ed H. Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, page 453–456, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605580111. doi: 10.1145/1357054.1357127. URL https://doi.org/10.1145/1357054.1357127.

M.L. Knapp, J.A. Hall, and T.G. Horgan. *Nonverbal Communication in Human Interaction*. Cengage Learning, 2013. ISBN 9781133311591. URL https://books.google.co.uk/books?id=-g7hkSR_mLoC.

Robert M. Krauss and Uri Hadar. 93The role of speech-related arm/hand gestures in word retrieval. In *Gesture, Speech, and Sign*. Oxford University Press, 07 1999. ISBN 9780198524519. doi: 10.1093/acprof:oso/9780198524519.003.0006. URL https://doi.org/10.1093/acprof:oso/9780198524519.003.0006.

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

Taras Kucherenko, Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, and Gustav Eje Henter. The GENEA Challenge 2020: Benchmarking gesture-generation systems on common data. In *Proceedings of the International Workshop on Generation and Evaluation of Non-VeItrbal Behaviour for Embodied Agents*, GENEA '20, 2020. URL https://genea-workshop.github.io/2020/.

Taras Kucherenko, Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, and Gustav Eje Henter. *A Large, Crowdsourced Evaluation of Gesture Generation Systems on Common Data: The GENEA Challenge 2020*, page 11–21. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450380171. URL https://doi.org/10.1145/3397481.3450692.

Takaaki Kuratate, Kevin G. Munhall, Philip E. Rubin, Eric Vatikiotis-Bateson, and Hani

Yehia. Audio-visual synthesis of talking faces from speech production correlates. In *Eurospeech'99*, volume 3, pages 1279–1282, 1999.

Zarrel V. Lambert and Richard M. Durand. Some precautions in using canonical analysis. *Journal of Marketing Research*, 12(4):468–475, 1975. doi: 10.1177/002224377501200411. URL https://doi.org/10.1177/002224377501200411.

Siddique Latif, Rajib Rana, Junaid Qadir, and Julien Epps. Variational Autoencoders for Learning Latent Representations of Speech Emotion: A Preliminary Study. In *Proc. Interspeech 2018*, pages 3107–3111, 2018. doi: 10.21437/Interspeech.2018-1568.

Siddique Latif, Rajib Rana, Sara Khalifa, Raja Jurdak, Junaid Qadir, and Björn Schuller. Survey of deep representation learning for speech emotion recognition. *IEEE Transactions on Affective Computing*, 14(2):1634–1654, 2023. doi: 10.1109/TAFFC.2021.3114365.

Binh H. Le, Xiaohan Ma, and Zhigang Deng. Live speech driven head-and-eye motion generators. *IEEE Transactions on Visualization and Computer Graphics*, 18(11):1902–1914, 2012. doi: 10.1109/TVCG.2012.74.

Lucian Leahu, Steve Schwenk, and Phoebe Sengers. Subjective objectivity: Negotiating emotional meaning. In *Proceedings of the 7th ACM Conference on Designing Interactive Systems*, DIS '08, page 425–434, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605580029. doi: 10.1145/1394445.1394491. URL https://doi.org/10.1145/1394445.1394491.

Willem J. M. Levelt. *Speaking: From Intention to Articulation*. The MIT Press, 08 1993. ISBN 9780262278225. doi: 10.7551/mitpress/6393.001.0001. URL https://doi.org/10.7551/mitpress/6393.001.0001.

J. Li, D. Kang, W. Pei, X. Zhe, Y. Zhang, Z. He, and L. Bao. Audio2gestures: Generating diverse gestures from speech audio with conditional variational autoencoders. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11273–11282, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society. doi: 10.1109/ICCV48922.2021.01110. URL https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.01110.

Y. Linde, A. Buzo, and R. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28(1):84–95, 1980. doi: 10.1109/TCOM.1980.1094577.

Erfan Loweimi, Peter Bell, and Steve Renals. On the robustness and training dynamics of raw waveform models. In *Proceedings of Interspeech 2020*, pages 1001–1005. International Speech Communication Association, October 2020. doi: 10.21437/Interspeech.2020-0017. Interspeech 2020, INTERSPEECH 2020 ; Conference date: 25-10-2020 Through 29-10-2020.

JinHong Lu and Hiroshi Shimodaira. A neural network based post-filter for speech-driven head motion synthesis. *arXiv e-prints*, art. arXiv:1907.10585, July 2019.

JinHong Lu and Hiroshi Shimodaira. Prediction of Head Motion from Speech Waveforms with a Canonical-Correlation-Constrained Autoencoder. pages 1301–1305,

2020. doi: 10.21437/Interspeech.2020-1218. URL http://dx.doi.org/10.21437/Interspeech.2020-1218. Proc. Interspeech 2020.

Hieu-Thi Luong, Shinji Takaki, Gustav Eje Henter, and Junichi Yamagishi. Adapting and controlling dnn-based speech synthesis using input codes. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4905–4909, 2017. doi: 10.1109/ICASSP.2017.7953089.

K.V. Mardia, J.T. Kent, J.T. Kent, and J.M. Bibby. *Multivariate Analysis*. Probability and Mathematical Statistics : a series of monographs and textbooks. Academic Press, 1979. ISBN 9780124712508. URL https://books.google.co.kr/books?id=bxjvAAAAMAAJ.

E.C. Marsi and F. van Rooden. Expressing uncertainty with a talking head in a multimodal question-answering system. In *Proceedings of the workshop on mulitmodal output generation (MOG 2007)*, pages 105–116. Centre for Telematics and Information Technology (CTIT), 2007. ISBN 15740846.

D. Matsumoto, M.G. Frank, and H.S. Hwang. *Nonverbal Communication: Science and Applications: Science and Applications*. EBSCO ebook academic collection. SAGE Publications, 2013. ISBN 9781412999304. URL https://books.google.co.uk/books?id=PeOeu3qFFTIC.

David Matsumoto and Hyisung C. Hwang. Evidence for training the ability to read microexpressions of emotion. *Motivation and Emotion*, 35:181–191, 2011.

Evelyn Z. McClave. Linguistic functions of head movements in the context of speech. *Journal of Pragmatics*, 32(7):855–878, 2000. ISSN 0378-2166. doi: https://doi.org/10.1016/S0378-2166(99)00079-X. URL https://www.sciencedirect.com/science/article/pii/S037821669900079X.

Gary McKeown, Michel Valstar, Roddy Cowie, Maja Pantic, and Marc Schroder. The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent. *IEEE Transactions on Affective Computing*, 3 (1):5–17, 2012. doi: 10.1109/T-AFFC.2011.20.

D. McNeill. *Hand and Mind: What Gestures Reveal about Thought*. Hand and Mind: What Gestures Reveal about Thought. University of Chicago Press, 1992. ISBN 9780226561325. URL https://books.google.co.kr/books?id=3ZZAfNumLvwC.

David Mcneill. So you do think gestures are nonverbal! reply to feyereisen (1987). *Psychological Review - PSYCHOL REV*, 94:499–504, 10 1987. doi: 10.1037/0033-295X.94.4.499.

David Mcneill. Hand and mind: What gestures reveal about thought. *Bibliovault OAI Repository, the University of Chicago Press*, 27, 06 1994. doi: 10.2307/1576015.

David McNeill. *Gesture and Thought*. University of Chicago Press, 2005.

Michael Meredith and Steve C. Maddock. Motion capture file formats explained. 2001. URL https://api.semanticscholar.org/CorpusID:64457840.

Abdelrahman Mohamed, Hung yi Lee, Lasse Borgholt, Jakob D. Havtorn, Joakim Edin, Christian Igel, Katrin Kirchhoff, Shang-Wen Li, Karen Livescu, Lars Maaloe, Tara N. Sainath, and Shinji Watanabe. Self-supervised speech representation learning: A review. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1179–1210, oct 2022. doi: 10.1109/jstsp.2022.3207050. URL https://doi.org/10.1109%2Fjstsp.2022.3207050.

Palmer Morrel-Samuels and Robert Krauss. Word familiarity predicts temporal asynchrony of hand gestures and speech. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18:615–622, 05 1992. doi: 10.1037/0278-7393.18.3.615.

D. Mortari, Michela Angelucci, and Landis Markley. Singularity and attitude estimation. *AAS 00-13, 10-th AIAA/AAS Space Flight Mechanics Meeting, Clearwaters, FL*, 01 2000.

K.G. Munhall, Jeffery Jones, Daniel Callan, Takaaki Kuratate, and Eric Vatikiotis-Bateson. Visual prosody and speech intelligibility head movement improves auditory speech perception. *Psychological science*, 15:133–7, 03 2004. doi: 10.1111/j.0963-7214.2004.01502010.x.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 689–696, Madison, WI, USA, 2011. Omnipress. ISBN 9781450306195.

Miriam A. Novack and Susan Goldin-Meadow. Gesture as representational action: A paper about function. *Psychonomic Bulletin & Review*, 24:652–665, 2017. URL https://api.semanticscholar.org/CorpusID:12943690.

Yoshiaki Ohkami. Spacecraft dynamics. In Robert A. Meyers, editor, *Encyclopedia of Physical Science and Technology (Third Edition)*, pages 431–448. Academic Press, New York, third edition edition, 2003. ISBN 978-0-12-227410-7. doi: https://doi.org/10.1016/B0-12-227410-5/00898-X. URL https://www.sciencedirect.com/science/article/pii/B012227410500898X.

Andrew J. Oxenham. How we hear: The perception and neural coding of sound. *Annual Review of Psychology*, 69:27–50, 2018.

Demet Özer and Tilbe Göksun. Gesture use and processing: A review on individual differences in cognitive resources. *Frontiers in Psychology*, 11, 2020. URL https://api.semanticscholar.org/CorpusID:226247618.

Sooha Park, Lee Jeremy, Jeremy Badler, and Norman Badler. Eyes alive. *ACM Transactions on Graphics (TOG)*, 21, 07 2002. doi: 10.1145/566570.566629.

M.L. Patterson. Nonverbal communication. In *Reference Module in Neuroscience and Biobehavioral Psychology*. Elsevier, 2017. ISBN 978-0-12-809324-5. doi: https://doi.org/10.1016/B978-0-12-809324-5.06502-0. URL https://www.sciencedirect.com/science/article/pii/B9780128093245065020.

Jonathan E. Peelle and Mitchell S. Sommers. Prediction and constraint in audiovisual speech perception. *Cortex*, 68:169–181, 2015. ISSN 0010-9452. doi: https:

//doi.org/10.1016/j.cortex.2015.03.006. URL https://www.sciencedirect.com/science/article/pii/S0010945215000957. Special issue: Prediction in speech and language processing.

David Peeters, Mingyuan Chu, Judith Holler, Peter Hagoort, and Aslı Özyürek. Electrophysiological and Kinematic Correlates of Communicative Intent in the Planning and Production of Pointing Gestures and Speech. *Journal of Cognitive Neuroscience*, 27(12):2352–2368, 12 2015. ISSN 0898-929X. doi: 10.1162/jocn_a_00865. URL https://doi.org/10.1162/jocn_a_00865.

Catherine Pelachaud, Norman I. Badler, and Mark Steedman. Generating facial expressions for speech. *Cognitive Science*, 20(1):1–46, 1996. doi: https://doi.org/10.1207/s15516709cog2001_1. URL https://onlinelibrary.wiley.com/doi/abs/10.1207/s15516709cog2001_1.

Jonas Persson, Kathryn Welsh, John Jonides, and Patricia Reuter-Lorenz. Cognitive fatigue of executive processes: Interaction between interference resolution tasks. *Neuropsychologia*, 45:1571–9, 05 2007. doi: 10.1016/j.neuropsychologia.2006.12.007.

David Poeppel and Xiangbin Teng. 2.06 - entrainment in human auditory cortex: Mechanism and functions. In Bernd Fritzsch, editor, *The Senses: A Comprehensive Reference (Second Edition)*, pages 63–76. Elsevier, Oxford, second edition edition, 2020. ISBN 978-0-12-805409-3. doi: https://doi.org/10.1016/B978-0-12-805408-6.00018-X. URL https://www.sciencedirect.com/science/article/pii/B978012805408600018X.

Amy S. Pollick and Frans B. M. de Waal. Ape gestures and language evolution. *Proceedings of the National Academy of Sciences*, 104(19):8184–8189, 2007. doi: 10.1073/pnas.0702624104. URL https://www.pnas.org/doi/abs/10.1073/pnas.0702624104.

Wim Pouw, Jacqueline A. de Nooijer, Tamara van Gog, Rolf A. Zwaan, and Fred Paas. Toward a more embedded/extended perspective on the cognitive function of gestures. *Frontiers in Psychology*, 5, 2014. URL https://api.semanticscholar.org/CorpusID:13970677.

Alexander S. Poznyak. 2 - rigid body kinematics. In Alexander S. Poznyak, editor, *Classical and Analytical Mechanics*, pages 31–88. Elsevier, 2021. ISBN 978-0-323-89816-4. doi: https://doi.org/10.1016/B978-0-32-389816-4.00013-2. URL https://www.sciencedirect.com/science/article/pii/B9780323898164000132.

Reza Sharif Razavian, Sara Greenberg, and John McPhee. Biomechanics imaging and analysis. In Roger Narayan, editor, *Encyclopedia of Biomedical Engineering*, pages 488–500. Elsevier, Oxford, 2019. ISBN 978-0-12-805144-3. doi: https://doi.org/10.1016/B978-0-12-801238-3.99961-6. URL https://www.sciencedirect.com/science/article/pii/B9780128012383999616.

Ulf-Dietrich Reips. Standards for internet-based experimenting. *Experimental psychology*, 49:243–56, 02 2002. doi: 10.1026//1618-3169.49.4.243.

Jörg Rett and Jorge Dias. Human robot interaction based on bayesian analysis of human

movements. volume 4874, pages 530–541, 12 2007. ISBN 978-3-540-77000-8. doi: 10.1007/978-3-540-77002-2_45.

R.E. Riggio and H.R. Riggio. Face and body in motion: Nonverbal communication. In Thomas Cash, editor, *Encyclopedia of Body Image and Human Appearance*, pages 425–430. Academic Press, Oxford, 2012. ISBN 978-0-12-384925-0. doi: https://doi.org/10.1016/B978-0-12-384925-0.00068-7. URL https://www.sciencedirect.com/science/article/pii/B9780123849250000687.

N. Sadoughi and C. Busso. Novel realizations of speech-driven head movements with generative adversarial networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018)*, 2018a.

Najmeh Sadoughi and Carlos Busso. Novel Realizations of Speech-Driven Head Movements with Generative Adversarial Networks. In *ICASSP*, pages 6169–6173, April 2018b. doi: 10.1109/ICASSP.2018.8461967.

Najmeh Sadoughi and Carlos Busso. Speech-driven animation with meaningful behaviors. *Speech Communication*, 110:90–100, 2019. ISSN 0167-6393. doi: https://doi.org/10.1016/j.specom.2019.04.005. URL https://www.sciencedirect.com/science/article/pii/S0167639318300013.

Hardik B. Sailor and Hemant A. Patil. Unsupervised Deep Auditory Model Using Stack of Convolutional RBMs for Speech Recognition. In *Proc. Interspeech 2016*, pages 3379–3383, 2016. doi: 10.21437/Interspeech.2016-812.

Tara N. Sainath, Ron J. Weiss, Andrew W. Senior, Kevin W. Wilson, and Oriol Vinyals. Learning the speech front-end with raw waveform CLDNNs. In *INTERSPEECH*, 2015.

M. Salem, F. Eyssel, Katharina J. Rohlfing, Stefan Kopp, and F. Joublin. To err is human(-like): Effects of robot gesture on perceived anthropomorphism and likability. *International Journal of Social Robotics*, 5:313–323, 2013.

Mehmet E. Sargin, Yucel Yemez, Engin Erzin, and Ahmet M. Tekalp. Analysis of head gesture and prosody patterns for prosody-driven head-gesture animation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(8):1330–1345, 2008. doi: 10.1109/TPAMI.2007.70797.

Mehmet Emre Sargin, YÜcel Yemez, Engin Erzin, and A. Murat Tekalp. Audiovisual synchronization and fusion using canonical correlation analysis. *IEEE Transactions on Multimedia*, 9(7):1396–1403, 2007. doi: 10.1109/TMM.2007.906583.

R. Schluter, I. Bezrukov, H. Wagner, and H. Ney. Gammatone features and feature combination for large vocabulary speech recognition. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 4, pages IV–649–IV–652, 2007. doi: 10.1109/ICASSP.2007.366996.

Felix Scholkmann, Jens Boss, and Martin Wolf. An efficient algorithm for automatic peak detection in noisy periodic and quasi-periodic signals. *Algorithms*, 5(4):588–603, 2012. ISSN 1999-4893. doi: 10.3390/a5040588. URL https://www.mdpi.com/1999-4893/5/4/588.

Jun'ichiro Seyama and Ruth S. Nagayama. The uncanny valley: Effect of realism on the impression of artificial human faces. *PRESENCE: Teleoperators and Virtual Environments*, 16:337–351, 2007.

Matt Shannon and William Byrne. Autoregressive HMMs for speech synthesis. In *Proc. Interspeech 2009*, pages 400–403, 2009. doi: 10.21437/Interspeech.2009-135.

Matt Shannon, Heiga Zen, and William Byrne. Autoregressive models for statistical parametric speech synthesis. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(3):587–597, 2013. doi: 10.1109/TASL.2012.2227740.

Ken Shoemake. Animating rotation with quaternion curves. *SIGGRAPH Comput. Graph.*, 19(3):245–254, jul 1985. ISSN 0097-8930. doi: 10.1145/325165.325242. URL https://doi.org/10.1145/325165.325242.

Inge Soderkvist and PA Wedin. Determining the movements of the skeleton using well-configured markers. *Journal of Biomechanics*, 26:1473–1477, January 1994. doi: 10.1016/0021-9290(93)90098-Y.

W. H. Sumby and Irwin Pollack. Visual contribution to speech intelligibility in noise. *The Journal of the Acoustical Society of America*, 26(2):212–215, 1954. doi: 10.1121/1.1907309. URL https://doi.org/10.1121/1.1907309.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA, 2014. MIT Press.

S. Takaki and J. Yamagishi. A deep auto-encoder based low-dimensional feature extraction from FFT spectral envelopes for statistical parametric speech synthesis. In *ICASSP*, pages 5535–5539, March 2016. doi: 10.1109/ICASSP.2016.7472736.

Kenta Takeuchi, Souichirou Kubota, Keisuke Suzuki, Dai Hasegawa, and Hiroshi Sakuta. Creating a gesture-speech dataset for speech-based automatic gesture generation. In Constantine Stephanidis, editor, *HCI International 2017 – Posters' Extended Abstracts*, pages 198–202, Cham, 2017. Springer International Publishing. ISBN 978-3-319-58750-9.

K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura. Speech parameter generation algorithms for hmm-based speech synthesis. In *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.00CH37100)*, volume 3, pages 1315–1318 vol.3, June 2000. doi: 10.1109/ICASSP.2000.861820.

Zoltán Tüske, Pavel Golik, Ralf Schlüter, and Hermann Ney. Acoustic modeling with deep neural networks using raw time signal for lvcsr. In *INTERSPEECH*, 2014.

Zoltán Tüske, Ralf Schlüter, and Hermann Ney. Acoustic Modeling of Speech Waveform Based on Multi-Resolution, Neural Network Signal Processing. In *ICASSP*, pages 4859–4863, 2018.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals,

Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. In *Proc. 9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, page 125, 2016.

Karl Van Orden, Tzyy-Ping Jung, and Scott Makeig. Combined eye activity measures accurately estimate changes in sustained visual task performance. *Biological Psychology*, 52:221–240, 05 2000. doi: 10.1016/S0301-0511(99)00043-5.

Virginie van Wassenhove, Ken W. Grant, David Poeppel, and Morris Halle. Visual speech speeds up the neural processing of auditory speech. *Proceedings of the National Academy of Sciences of the United States of America*, 102(4):1181–1186, 2005. ISSN 00278424. URL http://www.jstor.org/stable/3374398.

Peter Martin Vieting, Christoph Lüscher, Wilfried Michel, Ralf Schlüter, and Hermann Ney. On architectures and training for raw waveform feature extraction in asr. *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 267–274, 2021.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *J. Mach. Learn. Res.*, 11:3371–3408, December 2010. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1756006.1953039.

Alexei Vinokourov, Nello Cristianini, and John Shawe-Taylor. Inferring a semantic representation of text via cross-language correlation analysis. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002. URL https://proceedings.neurips.cc/paper/2002/file/d5e2fbef30a4eb668a203060ec8e5eef-Paper.pdf.

Ekaterina Volkova, Stephan de la Rosa, Heinrich H. Bülthoff, and Betty Mohler. The mpi emotional body expressions database for narrative scenarios. *PLOS ONE*, 9(12):1–28, 12 2014. doi: 10.1371/journal.pone.0113647. URL https://doi.org/10.1371/journal.pone.0113647.

Patrick von Platen, Chao Zhang, and Philip Woodland. Multi-Span Acoustic Modelling using Raw Waveform Signals. *ArXiv*, abs/1906.11047, 2019.

Petra Wagner, Zofia Malisz, and Stefan Kopp. Gesture and speech in interaction: An overview. *Speech Communication*, 57:209–232, 2014. ISSN 0167-6393. doi: https://doi.org/10.1016/j.specom.2013.09.008. URL https://www.sciencedirect.com/science/article/pii/S0167639313001295.

Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1083–1092. JMLR.org, 2015.

Steven Wegmann, Arlo Faria, Adam Janin, Korbinian Riedhammer, and Nelson Morgan. The tao of atwv: Probing the mysteries of keyword search performance. In *2013*

IEEE Workshop on Automatic Speech Recognition and Understanding, pages 192–197, 2013. doi: 10.1109/ASRU.2013.6707728.

E. Wigner. *Group Theory: And its Application to the Quantum Mechanics of Atomic Spectra*. Pure and applied physics. Elsevier Science, 2012. ISBN 9780323152785. URL https://books.google.com.hk/books?id=ENZzI49uZMcC.

Maria K. Wolters, Karl Isaac, and Steve Renals. Evaluating speech synthesis intelligibility using amazon mechanical turk. In Yoshinori Sagisaka and Keiichi Tokuda, editors, *The Seventh ISCA Tutorial and Research Workshop on Speech Synthesis, Kyoto, Japan, September 22-24, 2010*, pages 136–141. ISCA, 2010. URL http://www.isca-speech.org/archive/ssw7/ssw7_136.html.

Hani C. Yehia, Takaaki Kuratate, and Eric Vatikiotis-Bateson. Linking facial animation, head motion and speech acoustics. *Journal of Phonetics*, 30(3):555 – 568, 2002. ISSN 0095-4470. doi: https://doi.org/10.1006/jpho.2002.0165. URL http://www.sciencedirect.com/science/article/pii/S0095447002901658.

Y. Yoon, W. Ko, M. Jang, J. Lee, J. Kim, and G. Lee. Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4303–4309, 2019.

Takenori Yoshimura, Takato Fujimoto, Keiichiro Oura, and Keiichi Tokuda. SPTK4: An open-source software toolkit for speech signal processing. In *12th ISCA Workshop on Speech Synthesis (SSW 12)*, 2023.

Atef Ben Youssef, Hiroshi Shimodaira, and David A. Braude. Articulatory features for speech-driven head motion synthesis. In *INTERSPEECH*, 2013.

Dong Yu, Frank Seide, and Gang Li. Conversational speech transcription using context-dependent deep neural networks. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, page 1–2, Madison, WI, USA, 2012. Omnipress. ISBN 9781450312851.

Yu Zhang, William Chan, and Navdeep Jaitly. Very deep convolutional networks for end-to-end speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4845–4849, 2017. doi: 10.1109/ICASSP.2017.7953077.

Yu Zhou, Junfeng Li, Yanqing Sun, Jianping Zhang, Yonghong Yan, and Masato Akagi. A hybrid speech emotion recognition system based on spectral and prosodic features. *IEICE Trans. Inf. Syst.*, 93-D:2813–2821, 2010.

Sławomir K. Zieliński, Philip Hardisty, Christopher Hummersone, and Francis Rumsey. Potential biases in mushra listening tests. *Journal of The Audio Engineering Society*, 2007.

Elana Zion-Golumbic and Charles E. Schroeder. Attention modulates 'speech-tracking' at a cocktail party. *Trends in Cognitive Sciences*, 16(7):363–364, 2012. ISSN 1364-6613. doi: https://doi.org/10.1016/j.tics.2012.05.004. URL https://www.sciencedirect.com/science/article/pii/S1364661312001222.