Audio-driven Robot Upper-body Motion Synthesis

Jan Ondras, Oya Celiktutan, Paul Bremner and Hatice Gunes, *Senior Member, IEEE*

***Abstract*—Body language is an important aspect of human communication, which an effective human-robot interaction interface should mimic well. Human beings exchange information and convey their thoughts and feelings through gaze, facial expressions, body language and tone of voice along with spoken words, and infer 65% of the meaning of the communicated messages from these nonverbal cues. Modern robotic platforms are however limited in their ability to automatically generate behaviours that align with their speech. In this paper, we develop a neural network based system that takes audio from a user as an input and generates upper-body gestures including head, hand and torso movements of the user on a humanoid robot, namely, Softbank Robotics' Pepper. Our system was evaluated quantitatively as well as qualitatively using web-surveys when driven by natural speech and synthetic speech. We compare the impact of generic and person-specific neural network models on the quality of synthesised movements. We further investigate the relationships between quantitative and qualitative evaluations and examine how the speaker's personality traits affect the synthesised movements.**

***Index Terms*—audio-based motion generation; human-robot interaction; personality perception.**

I. Introduction

BODY language plays an important role in human communication. While speaking, people use facial expressions, head motion and hand gestures to convey the same meaning as speech and to complement and enrich the message [1]. Head movements contribute to speech comprehension [2], increase the level of perceived naturalness [3], warmth and competence [4], and also convey the emotional state of the speaker [5]. Likewise, hand and arm movements are significant for distinguishing between affective states [6]. It has been also shown that 90% of human gesticulation occurs while speaking [1] and that the speech and gestures originate in the same internal process and share the same semantic meaning [1, 7]. These results thus motivate the investigation of the correlation and synchrony between these two modalities. Voigt et al. [8] showed that there is a statistically significant correlation between prosodic features extracted from audio and raw body movements. Several studies [1, 9, 10] further confirmed the synchrony between gesture strokes and stressed syllables and also between gesture phrases and intermediate intonation phrases.

These abovementioned findings led to attempts at cross-modal prediction, namely, learning mapping from audio to movements. However, this problem is challenging as multiple audio signals can be associated with the same motion sequence or vice versa. In addition, humans tend to feel unsettled when humanoid robots do not act realistically [11], and when there is an incompatibility

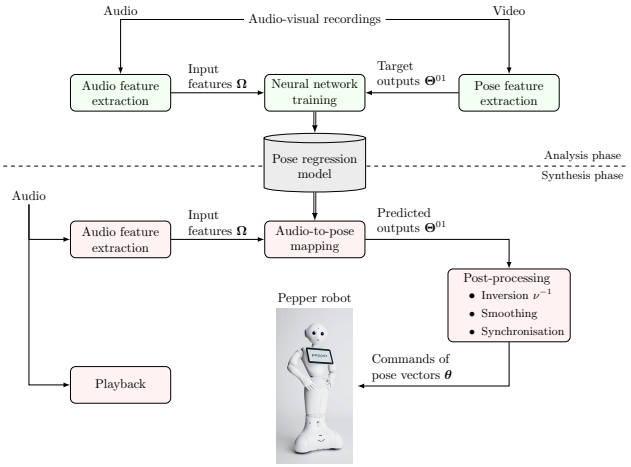


Fig. 1. Proposed audio-driven upper-body motion synthesis system. There are two main processing phases: (i) analysis phase (which happens offline); and (ii) synthesis phase (which can happen offline or online). The analysis phase involves extracting audio and pose features, and training pose regression model, namely learning the mapping between these features. In the synthesis phase, the trained pose regression model is applied to the audio features extracted from the audio input to predict the pose features, which are then post-processed to be displayed on the robot in real-time. A larger version of the figure is available in the supplementary material.

J. Ondras and H. Gunes are with the Department of Computer Science and Technology, University of Cambridge, Cambridge, UK CB3 0FD e-mail: jankondras@gmail.com, Hatice.Gunes@cl.cam.ac.uk.

O. Celiktutan is with the Centre for Robotics Research, Department of Engineering, King's College London, London, UK WC2R 2LS e-mail: oya.celiktutan@kcl.ac.uk.

P. Bremner is with the Bristol Robotics Laboratory, University of the West England, Bristol, UK BS16 1QY e-mail: paul.bremner@brl.ac.uk

between speech and associated gestures [12]. Several works developed audio-driven motion synthesis systems targeted at animated conversational agents and talking avatars, for example, for the synthesis of head motion [13–15] or hand movements [16, 17]. To the best of our knowledge, there has been no previous attempt at whole upper-body (head, hands and torso) motion synthesis from audio on a humanoid robot.

As shown in Fig. 1, this paper presents an automatic audio-driven upper-body motion synthesis system designed for the humanoid robot Pepper. The proposed system takes audio from a user as an input and generates upper-body motions in real-time. More explicitly, we extract a set of audio features and pose features using the dataset of [18]. This dataset contains audio-visual recordings captured while human participants were speaking about a topic, e.g., their hobbies, together with their self-assessments with respect to Big Five personality traits (e.g., extroversion, opennes, etc.). We then use multilayer perceptron (MLP) and long short-term memory (LSTM) neural network models to learn the mapping between audio features and pose features in a supervised manner. We compare two training approaches: (i) subject-independent (SI) - general model trained and evaluated on data from multiple subjects; and (ii) subject-dependent (SD) - specific model trained and evaluated on each subject separately. Finally, a filtering approach is proposed to smooth and constrain the synthesised motions to the robot's operating limits. The developed system[1] is extensively evaluated quantitatively in terms of various state-of-the-art metrics, and qualitatively by conducting a user study.

Overall, we argue that our contributions are: (i) we perform *whole upper-body* motion synthesis including head, hand and torso movements, unlike the previous works that synthesised either head or hand movements; (ii) we focus on a *humanoid robot*, differently from previous works that target motion synthesis on animated virtual characters or embodied agents; and (iii) we present a preliminary analysis on how personality of the human participants has an impact on the synthesised motions. Taking into account the potential use of humanoid robots in various application areas including education, healthcare, public spaces and much more, our work has significant implications for allowing more expressive and human-like communication using audio input only and hence enhancing user's satisfaction and robot's acceptance.

[1]Our implementation is available at https://github.com/jancio/Audio-driven-upper-body-motion-synthesis.

II. Related Work

In the area of audio-driven motion synthesis, early studies [26, 28, 29] usually relied on hard-coded rules deciding which motion pattern to synthesise. The major limitation of such rule-based systems is the repetitiveness of movements as there is only a fixed set of rules. Also, it is problematic to ensure synchronisation between verbal and non-verbal events using rules. Data-driven methods address these problems by capturing the variability of movements from the training data and implicitly learning the synchronisation between audio and motion. Previous works employed probabilistic models such as dynamic Bayesian networks (DBNs) [3], hidden Markov models (HMMs) [16, 30–33], conditional random fields (CRFs) [17] or Gaussian mixture models [34]. Recently, deep neural network models attained popularity in this field: Ding et al. [35] showed that deep neural networks generate better head motion sequences than HMMs. In particular, they used a multilayer perceptron (MLP) model and in their later work [19] they further improved the system with unsupervised pre-training using deep belief networks. The MLP models are however limited in modelling temporal data. Ding et al. [15] and Haag and Shimodaira [14] thus compared the MLP with a bidirectional long short-term memory (BLSTM) model in the head motion synthesis task. Both works reported improvement of the BLSTM-based system over the MLP-based one in terms of the naturalness of the synthesised motion assessed via a user study, root mean-squared error and canonical correlations between the original and synthesised head motion. Other studies experimented with generative models, for instance, Chiu et al. [27] used hierarchical factored conditional restricted Boltzmann machines (HFCRBMs) to generate hand movements. Greenwood et al. [20] introduced a generative head motion model based on the conditional variational autoencoder (CVAE) that allows prediction of several motion trajectories for the same audio. One major challenge of the data-driven methods is the lack of meaning - even if the generated movements are well synchronised with the speech they may lack or even contradict the meaning of the communicated message. Several studies [13, 22] thus developed hybrid approaches, for example, Sadoughi et al. [13] constrained their DBN model on several discourse functions (affirmation, negation, question, and backchannel).

Most related work focused only on head movements [13, 14, 19, 20] due to high correlations between speech and head motion, and some further studies included facial expressions [21, 24, 25]. Other related work [16, 17, 26, 27] synthesised hand movements exclusively. For example, Bozkurt et al. [16] used hidden semi Markov models (HSMMs) relying on the hierarchical model of hand

TABLE I
COMPARISON OF AUDIO-DRIVEN MOTION SYNTHESIS STUDIES SUMMARISED IN SECTION II.

Study	Target Domain	Method	Body Parts	3D Pose Source	Synthesis Mode	Recordings per Speaker (min)
[19]	avatar	MLP	head	single RGB cam.	offline	100
[13]	avatar	Hybrid DBN	head	mocap	offline	27
[20]	avatar	BLSTM, CVAE	head	3 RGB cams.	offline	180
[14]	avatar	BLSTM	head	mocap	offline	16
[15]	avatar	BLSTM	head	mocap	offline	263*
[21]	avatar	BLSTM	head, face	single RGB cam.	offline	146*
[22, 23]	avatar	Hybrid DBN	head, hands	mocap	offline	30, 66
[24, 25]	avatar	LSTM, GRU	head, face	single RGB cam.	online	6, 6
[16]	avatar	HSMM	hands	mocap, 4 RGB cams.	offline	20
[17]	avatar	CRF	hands	mocap	online	12
[26]	avatar	Rule-based	hands	mocap	offline	6
[27]	avatar	HFCRBM	hands	mocap	offline	1
Our work	robot	MLP, LSTM	head, hands, torso	single RGB cam.	online, offline	2

* denotes that duration of the recording was estimated based on the utterance duration of 12.5 seconds (typically 10-15 s).

gestures [9]. In closely related work, Sadoughi et al. [22] attempted to synthesise both head and hand movements simultaneously. The researchers focused on three prototypical hand gestures and two head gestures using two hybrid DBN models (one for head and one for hand gestures) which however required manual annotation of motion sequences. To the best of our knowledge, there was no previous work that performed whole upper-body motion synthesis, combining the synthesis of head, hand and torso movements in a fully automatic system.

A line of work [13–16, 19–22, 26, 27] developed offline systems that require the whole input audio upfront. In such cases there is no need for low-latency predictions and synthesis so that more complex models (e.g., BLSTM) can be used. Only Pham et al. [24, 25] used LSTM and also the gated recurrent unit (GRU) model for real-time head motion synthesis and facial animation, while Levine et al. [17] generated hand movements from live speech. Developing online (or real-time) systems has remained a less explored area.

Generating the appropriate social behaviours for robots is key to enhancing the user's interaction experience and their acceptance of the robot. Research on nonverbal cues has shown that head and body movements are significant predictors of personality [36]. Motivated by this, several works manually programmed the robot's movements to display either an extroverted personality or an introverted personality, and examined the effect of personality match (similar or complementary personality types) on the engagement state of the user during human-robot interactions [37, 38]. However, we are not aware of any previous work that investigates the personality perception of automatically generated behaviours nor performs personality-driven motion synthesis.

In Table I, we summarise the studies closely related to our work. Our work is the first system that enables whole upper-body motion synthesis from audio through combining head, hand and torso movements, aiming at a humanoid robot. We conduct a systematic study where we analyse the effectiveness of four different types of audio features (Section III-A1) and three different methods for estimating 3D pose from a single-view RGB video (Section III-A2). To learn the mapping between audio features and upper body pose, we train MLP and LSTM models following two approaches: (i) subject-independent (SI), resulting in a generic model; and (ii) subject-dependent (SD), resulting in a personalised model. We compare these models both quantitatively (in terms of three evaluation metrics) and qualitatively (via a user study). We present further results to (i) compare the qualitative preference for the MLP/LSTM model across natural speech and synthetic speech driven system; (ii) examine the relationships between quantitative and qualitative evaluation metrics for audio-driven robot motion synthesis; and (iii) investigate the impact of the speaker's personality traits on the synthesised robot movements.

III. AN AUDIO-DRIVEN UPPER-BODY MOTION SYNTHESIS SYSTEM

The architecture of the audio-driven upper-body motion synthesis system is shown in Fig. 1. There are two main processing phases: (i) analysis phase (which happens offline); and (ii) synthesis phase (which can happen offline or online). The analysis phase involves extracting audio and pose features, and training pose regression model, namely learning the mapping between these features. In the synthesis phase, the trained pose regression model is applied to the audio features extracted from the audio input to predict the pose features, which are then post-processed to be displayed on the robot in real-time.

A. Analysis Phase

1) Audio Feature Extraction: The state-of-the-art audio features are Mel frequency cepstral coefficients (MFCCs) together with log filter banks (LogFBs) that have recently become popular [15, 19, 20]. Another recent approach to audio feature extraction uses convolutional neural networks [24]. However, for speech recognition, it yields a performance on par with the hand-crafted MFCC/LogFB features under low levels of noise [39]. Therefore, in this paper we considered MFCC and LogFB features.

The stereo audio signal was first averaged to a single channel and downsampled to $f_s = 16$ kHz, similarly to [14, 15, 19, 20]. For each audio clip, we then extracted MFCC and LogFB features using the Python toolbox called python-speech-features [40]. In addition, we calculated differential features to represent the dynamic nature of the audio signal. Following [14, 15, 19], the first and second order time derivatives of the LogFB coefficients were appended to the LogFB feature vectors. Audio feature extraction process resulted in an audio feature set $\mathbf{\Omega} \in \mathbb{R}^{N_{fr} \times N_{fe}}$ where N_{fr} is the number of audio frames and N_{fe} is the size of each feature vector. For each audio frame i ($i = 1, \ldots, N_{fr}$), we extracted the following four audio feature sets: (i) MFCC-13 - 13 highest energy cepstral coefficients; (ii) LogFB-26 - 26 log filter bank coefficients; (iii) LogFB-52 - 26 log filter bank coefficients together with 26 their first-order differential derivatives; and (iv) LogFB-78 - 26 log filter bank coefficients together with their 26 first-order time derivatives and 26 second-order time derivatives. In all four cases we used the standard settings and z-normalisation per subject. The resulting rate of audio features was $f_f = 100$ Hz.

2) Pose Feature Extraction: For pose feature extraction, the first step was to estimate 3D human pose from a single-view raw images. This is an active research area with various methods being proposed. One approach is to first estimate 2D joint locations and then lift them to 3D. For instance, a 2D pose estimation framework such as OpenPose [41] can be combined with the 2D→3D matching approach of Chen and Ramanan [42]. Alternatively, there has been recent work that tries to directly predict 3D joint positions in real-time, e.g., VNect [43] and Lifting from the deep (LFTD) [44]. In order to choose the appropriate method for our task, we inspected these three approaches visually. We used LFTD method [44] for our task as it is more robust to missing/occluded joints and provides significantly less jerky trajectories as compared to the other two approaches.

In order to reproduce the upper-body pose on the humanoid robot, we calculated a set of 11 joint angles associated with head, shoulders, elbows, wrists and torso from the 3D pose estimated using LFTD method. The joint angles were originally calculated at the frame rate of $f_v = 50$ Hz, lower than the audio features ($f_f = 100$ Hz). To train a model between these two data streams, we synchronised these features in time by upsampling the joint angles trajectories using linear interpolation, similarly to [45]. The upper-body movements over N_{fr} time-steps are denoted by the matrix $\mathbf{\Theta} \in \mathbb{R}^{N_{fr} \times 11}$ of joint angles, henceforth.

We further post-processed the joint angle trajectories to mitigate the effect of noise. In particular, we smoothed the calculated joint angle trajectories as the LFTD method operates on a single image only without taking into account temporal dependencies. A similar approach was performed for the head motion synthesis in [19]. More explicitly, we applied the 5$^{\text{th}}$ order low-pass Butterworth filter with a cut-off frequency f_c. We set the cut-off frequency as $f_c = 4$ Hz through experimenting with several values commonly used in the related literature [46]. Finally, the smoothed joint angles were constrained to the robot's operating limits, and each joint angle was normalised to the range $[0, 1]$, resulting in the pose feature set $\mathbf{\Theta}^{01} = \nu(\mathbf{\Theta})$, where ν is the normalisation operator, and was used to train the pose regression model. These two post-processing steps ensured that all angles were treated equally during training, and also the predictions made by the pose regression model were implicitly constrained to the robot's limits.

B. Pose Regression Model

We performed time-continuous regression to learn the mapping from audio features $\mathbf{\Omega}$ to pose features $\mathbf{\Theta}^{01}$ using two methods, namely, Mutilayer Perceptron (MLP) and Long Short-Term Memory (LSTM) networks.

1) Multilayer Perceptron (MLP): Given an input feature vector $\mathbf{x} \in \mathbb{R}^{N_{fe}}$, the multilayer perceptron (MLP) model is trained to predict the output vector $\mathbf{y} \in \mathbb{R}^{N_o}$ of N_o predictions [47]. In our experiments, we set the number of input-layer units to the number of features N_{fe}, and trained a separate MLP for each audio feature set described in Section III-A1, where $N_{fe} = 13, 26, 52, 78$. We set the number of output-layer units N_o as the number of joint angles to be predicted, namely $N_o = 11$ in our case. We used the ReLU activation function for each hidden-layer and sigmoid for the output layer. The number of hidden layers and hidden units were optimised on a validation set as described in Section IV-A4. The detailed architecture of MLP model is also available in the supplementary material.

2) Long Short-Term Memory (LSTM): MLP treats each training example independently, and it does not exploit

Fig. 2. Sample snapshots from the audio-visual dataset used [18].

relations between instances adjacent in time within a sequence. In contrast, Long Short-Term Memory (LSTM) networks [48] directly model sequential data, incorporating past contexts using internal memory. In this paper, we particularly focused on the sequence labeling problem where given an input sequence $\mathbf{X} \in \mathbb{R}^{N_\tau \times N_{fe}}$ of N_τ time-steps with N_{fe} features per time-step, the task is to output the sequence $\mathbf{Y} \in \mathbb{R}^{N_\tau \times N_o}$ of the same length with N_o predictions per time-step. In case of variable-length sequences, the input sequence was first padded (usually with zero feature vectors) to the maximum sequence length N_τ. We segmented audio $\mathbf{\Omega}$ and pose $\mathbf{\Theta}^{01}$ features into sequences of N_τ frames with the stride of 1 frame. We detail how we selected N_τ and the optimum number of hidden layers/units for the synthesis phase in Section IV-A4. The detailed architecture of LSTM model is also available in the supplementary material.

3) Training: The neural network models were trained by minimising mean squared error loss function, which is typical for regression problems, $L = \frac{1}{N_o}\sum_{i=1}^{N_o}(\hat{y}_i - y_i)^2$, where $\hat{\mathbf{y}}, \mathbf{y} \in \mathbb{R}^{N_o}$ are the true and predicted output vectors, respectively. We used Adam for optimisation, and dropout regularisation to reduce overfitting.

C. Synthesis Phase

We developed two synthesis modes: (i) offline synthesis; and (ii) online synthesis. In the offline synthesis, the whole audio input was available upfront, and the prediction and synthesis of movements were performed offline. In the online synthesis, the movements were predicted and synthesised on-the-fly while the input audio was being captured and processed.

1) Offline Synthesis: First, we extracted audio features from the whole audio input using the method described in Section III-A1. The trained pose regression model was then applied to obtain the set of angle predictions $\mathbf{\Theta}^{01}$ corresponding to the whole audio input. We projected the normalised angles from the range of $[0, 1]$ to their original angle ranges via ν^{-1} (inverse normalisation), and the predicted angles were smoothed using the low-pass filter described in Section III-B, following the previous works [13, 14, 19, 27, 49]. The resulted pose vectors were sent to the robot to perform the target movements. To synchronise with the audio playback running in parallel, these commands were triggered at the original feature frame rate $f_f = 100$ Hz by injecting an adaptive sleep time between two synthesised outputs.

2) Online Synthesis: We captured the audio in real-time using pyaudio library [50], with the same settings as the audio used for model training, namely, single channel and $f_s = 16$ kHz sampling rate. At each time-step (clocked at $f_f = 100$ Hz), the most recent $wf_s = 400$ audio samples within an audio feature extraction window w were used to calculate a single audio feature vector. This feature vector was then z-normalised, and was fed into the pose regression model as in the offline mode. Differently from the offline mode, smoothing was performed using the Kalman filter, as it is commonly applied in real-time motion synthesis systems [46, 51]. The pose vectors were finally sent to the robot, and the adaptive sleep time was applied as in the offline mode.

IV. Quantitative Results

A. Experimental Setup

1) Dataset: We used the audio-visual dataset introduced in [18], where upper-body gesturing of 20 subjects was captured while talking. Sample snapshots from the recordings are provided in Fig. 2. Each subject was asked to perform two tasks: (i) describe one of their hobbies, and (ii) tell a dramatic story based on a given picture. In total, 40 videos (one for each task) were recorded using an RGB camera with a video frame rate $f_v = 50$ Hz and stereo audio sampled at frequency $f_a = 44.1$ kHz. For each subject the dataset further contains their self-reported personality along the Big Five personality traits: *Extroversion*, *Agreeableness*, *Conscientiousness*, *Neuroticism* and *Openness*. Each personality trait was measured on 1–10 Likert scale using the Big Five Inventory test (BFI-10) [52].

2) Robotic Platform: As a robotic platform, we used the humanoid robot Pepper (version 1.7 and body type V16) developed by SoftBank Robotics [53]. The Pepper robot is 1.20 m tall wheeled humanoid robot with 17 joints and three omni-directional wheels.

We controlled the following 11 joint angles of the robot's upper body: HeadPitch, HeadYaw, LShoulderRoll, LShoulderPitch, LElbowRoll, LElbowYaw, RShoulderRoll, RShoulderPitch, RElbowRoll, RElbowYaw, and HipRoll. The hip pitch and knee pitch angles were always kept at their default values to ensure standing robot pose. Wrist yaw angles were also set to their default values as these angles do not contribute to gesturing and it was not possible to estimate these angles from videos adequately.

TABLE II
BEST MLP-SI MODEL ARCHITECTURE FOR EACH AUDIO FEATURE SET. N_l^* AND N_u^* DENOTE THE OPTIMAL NUMBER OF HIDDEN LAYERS AND HIDDEN UNITS PER HIDDEN LAYER, RESPECTIVELY.

Feature	$(N_l^*,\ N_u^*)$	RMSE	$\text{LCCA}_{\boldsymbol{\Theta}}$
MFCC-13	(5,8)	13.66	0.9766
LogFB-26	(7,8)	13.52	0.9765
LogFB-52	(5,16)	13.44	0.9756
LogFB-78	(7,16)	13.73	0.9751

To speedup the development and testing of the proposed system we simulated the virtual robot in the Choregraphe environment [54]. The system was implemented using NaoQi framework (version 2.5.5) and Python SDK provided by SoftBank Robotics. The robot was controlled by sending commands specifying the joint angles at desired times. Prior to this, each angle was limited by its lower bound and upper bound as provided for Pepper in [55].

3) Evaluation Metrics: Due to the *many-to-many mapping* nature of the problem, there are multiple equally correct answers and there is no single quantitative metric that could be relied on [20]. We therefore assessed the system's viability in terms of three evaluation metrics including root mean squared error (RMSE), local canonical correlation analysis (CCA), and jerkiness (J). We briefly define these metrics below, where we denote the input matrix of audio features by $\boldsymbol{\Omega} \in \mathbb{R}^{N_{fr} \times N_{fe}}$, the matrix of ground truth joint angles by $\boldsymbol{\Theta}^t \in \mathbb{R}^{N_{fr} \times 11}$, and the output matrix of predicted joint angles by $\boldsymbol{\Theta}^p \in \mathbb{R}^{N_{fr} \times 11}$. N_{fr} and f_r are the number of frames and the prediction frame rate. The RMSE over N_{fr} frames for all joint angles $\{\theta_j\}_{j=1:11}$ is given by $\text{RMSE}(\theta_j) = [\frac{1}{N_{fr}} \sum_{i=1}^{N_{fr}} \left(\boldsymbol{\Theta}_{i,j}^t - \boldsymbol{\Theta}_{i,j}^p\right)^2]^{0.5}$.

As a complimentary metric, we used the canonical correlation analysis (CCA) [56]. Contrary to standard correlation, CCA can operate on multi-dimensional data rather than on single column vectors. Since the evaluated motion sequences consist of several shorter gestures, the linear correlations evaluated by CCA are unlikely to hold at longer time-scales (e.g. the whole video). In such cases, CCA is calculated using sliding-window approach with the window size of N_τ datapoints and 1 datapoint stride. This is also known as local CCA (LCCA). LCCA allows comparisons of movements with different lengths (i.e. long vs short videos). Considering the hand gesture durations reported in previous works (0.3-5 s [49] and 2.49 s [57]) and also the choice of 3 s windows to capture distinct head movements in the related speech-driven head motion synthesis study [14], we set the time window to $\tau = 3$ s (i.e., $N_\tau = \tau f_f$ frames). We used LCCA to calculate correlation between true and predicted movements, $\text{LCCA}_{\boldsymbol{\Theta}} = \text{LCCA}(\boldsymbol{\Theta}^t, \boldsymbol{\Theta}^p)$.

Last metric to gauge the quality of the synthesised motion is angular jerkiness [16]. Overall angular jerkiness J of the motion trajectory is defined as

$$J(\boldsymbol{\Theta}) = \frac{1}{2f_f(N_{fr}-3)} \sum_{i=1}^{N_{fr}-3} \sum_{j=1}^{11} \left(\Delta^3 \boldsymbol{\Theta}_{i,j}\right)^2 \quad (1)$$

where $\Delta^3 \boldsymbol{\Theta}_{i,j} = (\boldsymbol{\Theta}_{i+3,j} - 3\boldsymbol{\Theta}_{i+2,j} + 3\boldsymbol{\Theta}_{i+1,j} - \boldsymbol{\Theta}_{i,j}) f_f^3$ is the 3rd-order forward difference of angle θ_j. If the jerkiness $J(\boldsymbol{\Theta}^t)$ of the ground truth motion is known, the absolute difference $\Delta J = |J(\boldsymbol{\Theta}^t) - J(\boldsymbol{\Theta}^p)|$ can be used to compare the smoothness of the generated sequence against the ground truth. The overall jerkiness over n motion sequences is simply calculated by taking the mean, $\Delta\tilde{J} = \frac{1}{n} \sum_{i=1}^{n} \Delta J_i$.

Since we have unequal sample sizes and unequal variance, we opted for Welch's t-test and Kolmogorov-Smirnov test to evaluate significance of both quantitative results and qualitative results. Prior to this, we first performed Anderson-Darling Normality test for each set to be compared (e.g., MLP-SI vs. MLP-SD) separately. If both sets passed the Normality test (i.e. the set of samples were very likely to come from the Gaussian distribution), Welch's t-test was applied to assess whether the difference between the two sets was statistically significant or not. Otherwise, Kolmogorov-Smirnov test was used with the same threshold p-value (i.e. either $p = 0.001$ or $p = 0.05$).

4) Implementation Details: We implemented both MLP and LSTM models in Python using the machine learning library Keras. Both models were trained using two different approaches: (i) subject-independent (SI) - generic model trained and evaluated on data from multiple subjects; and (ii) subject-dependent (SD) - specific model trained and evaluated on each subject separately. In both cases, the dataset was split into training (30 videos, 15 subjects), validation (4 videos, 2 subjects) and test (4 videos, 2 subjects) partitions. MLP and LSTM, each trained by two different approaches, will be referred to as MLP-SI, LSTM-SI, MLP-SD and LSTM-SD, henceforth.

We optimised the model architecture and selected the best audio feature set for MLP-SI model only, and trained the remaining models using the same approach. We searched for the best model architecture in terms of the number of hidden layers, $N_l \in \{1, 2, 3, 5, 7\}$, and the number of units, $N_u \in \{2^3, 2^4, \ldots, 2^9\}$, per hidden layer (same for all hidden layers) by following previous works [14, 15]. For each combination (N_l, N_u) of these hyperparameters, a separate model was trained for each audio feature set $\boldsymbol{\Omega}$ and evaluated on the validation

set. Using the best model architecture for each feature set, we performed 10-fold subject-independent cross-validation to compare the four audio feature sets. Each fold comprised 2 subjects for validation, and 15 subjects for training. The results averaged over all test subjects are summarised in Table II in terms of RMSE and $\text{LCCA}_{\mathbf{\Theta}}$, together with the best hyperparameters found for each feature set. As compared to the existing MLP-based audio-driven motion synthesis systems [14, 15], the obtained optimal architectures have fewer parameters, which correctly reflects the use of a smaller dataset.

Looking at Table II, all feature sets performed similarly in terms of each evaluation metric. Since the difference was negligible, we chose LogFB-26 audio features set for the rest of our experiments to keep the computational complexity low. MLP-SD models were then trained for each subject, using LogFB-26 features and its corresponding optimal architecture $(N_l = 7, N_u = 8)$.

Similarly, we determined the optimal architecture for LSTM-SI model using the chosen audio feature set, namely, LogFB-26. We segmented the dataset (both audio $\mathbf{\Omega}$ and pose $\mathbf{\Theta}^{01}$ feature sets in the same way) into sequences of N_τ frames with the stride of 1 frame. Based on the previous works [49, 57], we set $N_\tau = 300$ frames (corresponding to 3 s at frame rate $f_f = 100$ Hz) so that a whole gesture can be captured within one sequence used by LSTM. We chose one hidden layer for the LSTM model as it yielded results on par with the optimal architecture for MLP, and searched for the optimal number of hidden units per hidden layer, namely, $N_{u'} \in \{3 : 3 : 27\}$. For each hyperparameter $N_{u'}$ a separate model was trained for a maximum of 100 epochs with early stopping with the window size of 10, and the training batch size was set to 15,000 sequences. Each model was then evaluated on the validation set. We set $N_{u'} = 12$ as it resulted in the minimum validation loss. Finally, the LSTM-SD models were trained for each subject using the optimal architecture ($N_{u'} = 12$). The training settings were the same as for the LSTM-SI except that the maximum number of epochs was increased to 500 as the training of SD models did not converge as fast as the SI variant, and each training batch contained all training sequences.

B. Experimental Results

1) Model comparison - SI vs SD and MLP vs LSTM: We compared all four models, namely, MLP-SI, LSTM-SI, MLP-SD, and LSTM-SD, quantitatively in terms of the evaluation metrics introduced in Section IV-A3 as well as real-time synthesis latency. All four models were evaluated on the (unseen) test partition, and the SI models were trained and tested using 10-fold protocol so that the model was evaluated for each subject. The results for each model are compared in Table III. In general, SD approach performed better than SI in terms of $\text{LCCA}_{\mathbf{\Theta}}$ and $\Delta\tilde{J}$, and significantly better (p-value < 0.05 denoted by †) in terms of RMSE. Fig. 3-(a) further compares each model in terms of RMSE for each body joint. Overall, shoulder and elbow related angles yielded larger RMSE values, suggesting that these angles are more challenging to predict accurately.

TABLE III
QUANTITATIVE MODEL COMPARISON ON THE TEST SET. * DENOTES SIGNIFICANTLY BETTER PERFORMANCE (p-VALUE < 0.001) BETWEEN MLP AND LSTM MODEL FOR EACH VARIANT (SI/SD) SEPARATELY. † DENOTES SIGNIFICANTLY BETTER PERFORMANCE (p-VALUE < 0.05) BETWEEN THE SI AND SD VARIANT OF THE SAME MODEL TYPE (MLP/LSTM).

Model	RMSE	$\text{LCCA}_{\mathbf{\Theta}}$	$\Delta\tilde{J}$
MLP-SI	13.71	0.9757	1.333
LSTM-SI	13.70	0.9817*	1.330
MLP-SD	$9.17^\dagger$	0.9760	1.019
LSTM-SD	$10.23^\dagger$	0.9818*	1.016

Table III shows that the difference between MLP and LSTM models was relatively small for either variant, SI or SD. Specifically, for RMSE and $\Delta\tilde{J}$ measures the differences between MLP and LSTM were not found to be statistically significant (p-value > 0.05), whereas the difference was negligible for $\text{LCCA}_{\mathbf{\Theta}}$ values. In summary, the quantitative evaluation showed that SD variants performed better than SI, resulting in significantly smaller RMSE values, smaller $\Delta\tilde{J}$ values and slightly larger $\text{LCCA}_{\mathbf{\Theta}}$. However, it did not indicate MLP was consistently better than LSTM, or vice versa.

2) Real-time synthesis latency: We further evaluated the latency of motion synthesis through MLP and LSTM models operating in the online synthesis mode (Section III-C2). In particular, we measured the model inference latencies to be $\tau_{MLP} = 1 \pm 1$ ms and $\tau_{LSTM} = 41 \pm 10$ ms of the MLP-SI and LSTM-SI model respectively, over 10,000 inferences. We also obtained the latency of all other per-frame operations (i.e., reading the audio stream, z-normalisation, Kalman filtering, and dispatching the commands to the robot) as $\tau_{ops} = 9 \pm 6$ ms, same for each model type. All the measurements were made using a quad-core 2.2GHz Intel i7 CPU and 8GB RAM. Comparing the overall per-frame processing latencies $\tau_{MLP} + \tau_{ops}$ and $\tau_{LSTM} + \tau_{ops}$, we found that MLP model can perform the online motion synthesis 5-times faster than LSTM approximately. Although the latency of other per-frame operations τ_{ops} is significantly higher than the MLP

inference latency τ_{MLP} (p-value < 0.001), the overall MLP per-frame processing latency $\tau_{MLP} + \tau_{ops}$ is comparable with the frame period $\tau_f = f_f^{-1} = 10$ ms used to develop the models. This suggests that the MLP model is more suitable for real-time motion synthesis. If the LSTM model has to be used, the movements should be synthesised at about 5-times slower rate ($\sim$ 20 Hz), which is reasonable as humans begin to perceive series of images as a motion at around 15 images per second [58].

V. Qualitative Results

A. Experimental Setup

The developed system was qualitatively evaluated via a user study to (i) qualitatively compare MLP and LSTM models; (ii) compare the behaviour of the system driven by natural speech versus synthetic speech; and (iii) investigate how the speaker's personality traits affect the perceived appropriateness of the synthesised movements for the speech. To do so, we created two surveys: (i) natural speech based survey; and (ii) synthetic speech based survey.

For the evaluation based on natural speech, we used audio recordings from the original dataset [18]. The movements of the Pepper robot generated using each model (MLP-SI, LSTM-SI, MLP-SD, LSTM-SD) were recorded in the offline synthesis mode using the test audio clips. This resulted in 152 short audio-visual clips (4 models $\times$ 38 videos), each of less than 15 s. For each of the 38 original audio recordings, we then created two side-by-side videos: MLP-SI vs. LSTM-SI and MLP-SD vs. LSTM-SD. All video pairs (76 in total) were randomised and included in the survey. Prior to taking the survey, the participants were informed about which joints of the Pepper robot were controlled and would be assessed. They were then asked to view each video pair and assess each video with respect to the appropriateness of the generated movements for the given audio on a Likert scale ranging from very inappropriate (1) to very appropriate (5).

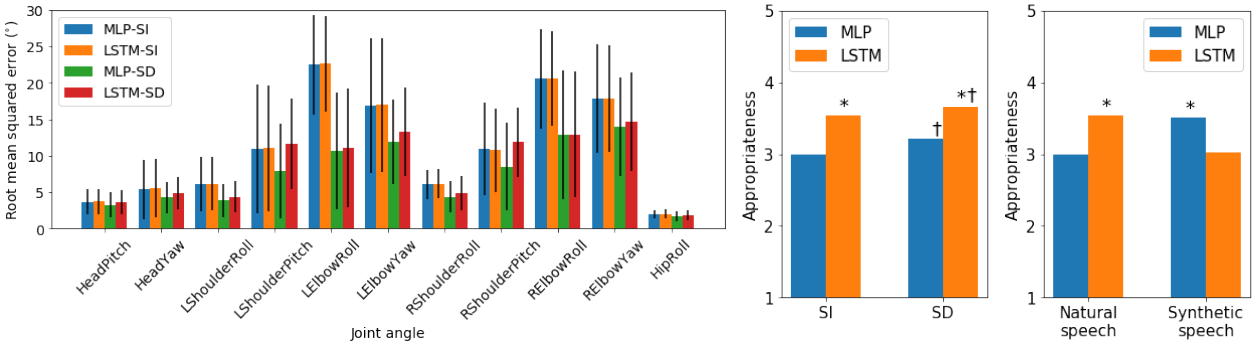
For the evaluation based on synthetic speech, we collated four short stories from the Strange Stories [59]. We then used the open-source text-to-speech system MaryTTS 5.2 [60] to create synthetic audio recordings. The MaryTTS system provides four voice types, each represented by one character with a specific personality type [61], namely, Obadiah (male, gloomy and depressed), Spike (male, angry and argumentative), Prudence (female, pragmatic and practical), and Poppy (female, outgoing and optimistic). To capture larger variety of speaking styles, we synthesised speech by each of these characters for every story, with further MaryTTS settings of HMM-based models and Great Britain English accent. This resulted in the set of 16 audio recordings of less than 40 s each. Analogously to the natural speech based survey, we created 16 video pairs to compare MLP-SI and LSTM-SI models. Note that in this case it was not possible to use the SD models as none of them was trained on none of the character's data. The participants were asked to perform the same task as in the natural speech based survey. After eliminating the unreliable responses, we obtained 20 and 43 responses for the natural and synthetic speech based survey, respectively.

B. Experimental Results

1) Model comparison - SI vs SD and MLP vs LSTM: The comparison of models based on the results from the natural speech based survey is shown in Fig. 3-(b). The movements synthesised by the SD variant were assessed as significantly more appropriate for the audio than those by the SI variant (p-value < 0.001 denoted by †). This is in line with the quantitative results in Section IV-B1, namely, subject-specific models were found to be better for this task.

Fig. 3-(b) further shows that the movements generated by the LSTM model were considered as significantly more appropriate for the audio than those by the MLP model (p-value < 0.001 denoted by *). This indicates that even though there were no clear quantitative differences between the two models, humans perceive considerable differences between them. These results also reproduced the findings in previous works [14, 15], showing that LSTM-based models outperformed MLP models in the head motion synthesis tasks, for whole upper-body motion synthesis.

2) Natural vs synthetic speech: For the SI model variant, the preference for the LSTM model was compared with the results from the synthetic speech based survey. As can be seen from Fig. 3-(c), in the case of synthetic speech the movements generated by the MLP model were assessed as more appropriate than those by the LSTM, which directly contradicts the results based on the natural speech. This can be explained by the combination of the following three facts: (i) the synthetic speech is more machine-like and choppy than the natural speech as the changes in intonation and stress patterns are not yet fully understood [62], it further lacks the natural phonetic variability [63, 64] and it is also less intelligible than the natural speech [64]; (ii) MLP model generates more machine-like and less smooth movements than LSTM as each frame is treated independently, which was also validated by the results in Section V-B1 and previous works [14, 15]; and (iii) the participants were asked to assess how appropriate the movements were for the



(a) (b) (c)

Fig. 3. (a) RMSE averaged over subjects for each joint angle, for each model evaluated on test set. (b) Appropriateness of the generated movements for the given audio, assessed in the natural speech based survey. * denotes significantly higher ratings (p-value < 0.001) between MLP and LSTM models, for each variant (SI/SD). † denotes significantly higher ratings (p-value < 0.001) between SI and SD variants of the same model type (MLP/LSTM). (c) Appropriateness of the generated movements for the given audio, assessed in natural and synthetic speech based surveys, comparing the SI variants of the MLP and LSTM models. * denotes significantly higher ratings (p-value < 0.001) between MLP and LSTM models, separately for natural and synthetic speech based evaluation.

speech. As shown in Fig. 3-(c), they rated the *machine-like speech* to match the *machine-like movements* better than the more natural movements generated by LSTM.

VI. Relationship to Personality

We examined the relationship between speakers' personality traits and generated movements in terms of quantitative and qualitative evaluation metrics for each neural network model type. We particularly investigated natural speech based motion synthesis using the audio-visual dataset [18].

For natural speech based motion synthesis, given the subjects' self-assessments within the range of 1-10, we first calculated the mean over all subjects and then categorised all 19 subjects into low and high classes based on the calculated mean for each personality trait. For each trait, we then evaluated the differences between low/high personality trait category along the four evaluation metrics including RMSE, LCCA$_{\mathbf{\Theta}}$, $\Delta\tilde{J}$ and appropriateness survey responses. This was done for each model type (i.e., MLP-SI, MLP-SD, LSTM-SI, and LSTM-SD) separately. We tested the statistical significance of the difference between low and high personality traits for a particular evaluation metric by following the same procedure as described in the last paragraph of Section IV-A3. Out of all 80 comparisons (pairwise low vs. high personality trait: 5 personality traits $\times$ 4 evaluation measures $\times$ 4 model types), we obtained statistically significant differences (p-value < 0.05) between low and high conscientiousness trait categories along the (i) LCCA$_{\mathbf{\Theta}}$ metric (with higher LCCA$_{\mathbf{\Theta}}$ associated with higher conscientiousness), and (ii) appropriateness measure (with higher appropriateness associated with higher conscientiousness). Both of these differences were found in the case of the MLP-SD model.

Even though none of the models was developed with explicitly providing the information about subjects' personalities, the MLP-SD model implicitly learned these relationships for conscientiousness. We can thus conclude that the MLP-SD model generates movements that are more correlated with the ground truth movements and are also assessed as more appropriate to the input audio for more conscientious people (better organised, more reliable) than for less conscientious. These relationships constitute a stepping stone towards learning to synthesise motions for distinctive personality styles rather than manually manipulating robot's behaviours [37, 38], which remains an open research problem in the field.

VII. Discussion

As demonstrated in Table I, each existing work differs in terms of its target domain (avatar vs. robot), the target body parts (head, hands, face), and the 3D pose source. Moreover, each work employed different sets of evaluation metrics, and collected and used their own in-house dataset. Therefore, a direct and quantitative comparison between various works in the field is not feasible. In addition, there is a lack of standardised multimodal corpora and standardised evaluation metrics agreed upon and used by all research groups.

Nevertheless, we compared our method with existing work with respect to the inference latency. Out of all related works presented in Section II, only Pham et al. [25] reported latency measurements. They measured an inference latency of 5 ms for their real-time LSTM-based head and facial movements synthesis sytem, which

is considerably smaller than the latency $\tau_{LSTM} = 41 \pm 10$ ms. However, unlike our work, their predictions were made using GPU which significantly lowers the inference latency.

VIII. Conclusion

Looking at our quantitative and qualitative results, we can draw the following conclusions. SD model variants outperform SI for both MLP and LSTM model types, suggesting that it is best to develop subject-specific models for this task. Although this might limit the scalability of the proposed approach, we argue that personalised movement generation increases the naturalness of the generated movements (validated by jerkiness measure), enables robots to express themselves with different styles based on varying contexts, and hence enhances user's satisfaction and robot's acceptance.

Quantitative comparison of the MLP and LSTM model did not clearly show which one generates better movements. Although both model types are suitable for real-time motion synthesis, the MLP model enables approximately 5-times faster synthesis as compared to the LSTM model. On natural speech, the movements generated by the LSTM model were assessed as significantly more appropriate for the given audio than those generated by the MLP model and this was the case for both SI and SD model variants. This result generalises the findings of previous speech-to-head-motion works to the whole upper-body motion synthesis. On synthetic speech, the survey respondents preferred the MLP model over LSTM, which reflects the fact that the more machine-like movements generated by the MLP model better match the more machine-like synthetic speech.

The analysis of the relationship between the human speaker's personality and the generated motion is one of the major features that makes our work distinctive. To the best of our knowledge, this is the first work to look at audio-to-motion synthesis from this perspective. We found several significant relationships between the speaker's personality traits and the motion synthesised for the speaker. However, to generate behaviours reflecting a particular personality type, personality-specific models need to be developed.

Finally, the work presented in this paper can be extended further by taking into account the semantics of the speech signals. The synthesised movements may be well synchronised with the speech, however they may be uncorrelated with the meaning of the messages being communicated or even contradict the meaning (e.g. head nodding for disagreement). Following hybrid speech-to-head-motion models [13, 22], a set of dialogues acts can be extracted from text using supervised classifiers and then used to constrain the models to generate movements from a particular category of motion patterns.

Acknowledgment

Jan Ondras conducted this work while studying at the University of Cambridge. Hatice Gunes' research is supported by the EPSRC under Grant Ref.: EP/R030782/1. The dataset used in this study has been collected as part of the EPSRC IDEAS Factory Sandpits call on Digital Personhood (Grant Ref.: EP/L00416X/1).

References

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

[2] K. G. Munhall, J. A. Jones, D. E. Callan, T. Kuratate, and E. Vatikiotis-Bateson, "Visual prosody and speech intelligibility: Head movement improves auditory speech perception," *Psychological science*, vol. 15, no. 2, pp. 133–137, 2004.

[3] S. Mariooryad and C. Busso, "Generating human-like behaviors using joint, speech-driven models for conversational agents," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 8, pp. 2329–2340, 2012.

[4] H. Van Welbergen, Y. Ding, K. Sattler, C. Pelachaud, and S. Kopp, "Real-time visual prosody for interactive virtual agents," in *International Conference on Intelligent Virtual Agents*. Springer, 2015, pp. 139–151.

[5] C. Busso, Z. Deng, U. Neumann, and S. Narayanan, "Learning expressive human-like head motion sequences from speech," in *Data-Driven 3D Facial Animation*. Springer, 2008, pp. 113–131.

[6] Z. Yang, A. Metallinou, E. Erzin, and S. Narayanan, "Analysis of interaction attitudes using data-driven hand gesture phrases," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 699–703.

[7] J. Cassell, D. McNeill, and K.-E. McCullough, "Speech-gesture mismatches: Evidence for one underlying representation of linguistic and nonlinguistic information," *Pragmatics & cognition*, vol. 7, no. 1, pp. 1–34, 1999.

[8] R. Voigt, R. J. Podesva, and D. Jurafsky, "Speaker movement correlates with prosodic indicators of engagement," in *Speech Prosody*, vol. 7, 2014.

[9] A. Kendon, "Gesticulation and speech: Two aspects of the process of utterance," *The relationship of verbal and nonverbal communication*, vol. 25, no. 1980, pp. 207–227, 1980.

[10] D. P. Loehr, "Temporal, structural, and pragmatic synchrony between intonation and gesture," *Laboratory Phonology*, vol. 3, no. 1, pp. 71–89, 2012.

[11] M. Mori, "The uncanny valley," *Energy*, vol. 7, no. 4, pp. 33–35, 1970.

[12] C. Ennis, R. McDonnell, and C. O'Sullivan, "Seeing is believing: body motion dominates in multisensory conversations," in *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4. ACM, 2010, p. 91.

[13] N. Sadoughi, Y. Liu, and C. Busso, "Meaningful head movements driven by emotional synthetic speech," *Speech Communication*, vol. 95, pp. 87–99, 2017.

[14] K. Haag and H. Shimodaira, "Bidirectional lstm networks employing stacked bottleneck features for expressive speech-driven head motion synthesis," in *International Conference on Intelligent Virtual Agents*. Springer, 2016, pp. 198–207.

[15] C. Ding, P. Zhu, and L. Xie, "Blstm neural networks for speech driven head motion synthesis," in *16th Annual Conference of the International Speech Communication Association*, 2015.

[16] E. Bozkurt, Y. Yemez, and E. Erzin, "Multimodal analysis of speech and arm motion for prosody-driven synthesis of beat gestures," *Speech Communication*, vol. 85, pp. 29–42, 2016.

[17] S. Levine, P. Krähenbühl, S. Thrun, and V. Koltun, "Gesture controllers," in *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4. ACM, 2010, p. 124.

[18] P. Bremner, O. Celiktutan, and H. Gunes, "Personality perception of robot avatar tele-operators," in *11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2016, pp. 141–148.

[19] C. Ding, L. Xie, and P. Zhu, "Head motion synthesis from speech using deep neural networks," *Multimedia Tools and Applications*, vol. 74, no. 22, pp. 9871–9888, 2015.

[20] D. Greenwood, S. Laycock, and I. Matthews, "Predicting head pose in dyadic conversation," in *International Conference on Intelligent Virtual Agents*. Springer, 2017, pp. 160–169.

[21] X. Lan, X. Li, Y. Ning, Z. Wu, H. Meng, J. Jia, and L. Cai, "Low level descriptors based dblstm bottleneck feature for speech driven talking avatar," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 5550–5554.

[22] N. Sadoughi and C. Busso, "Speech-driven animation with meaningful behaviors," *arXiv preprint arXiv:1708.01640*, 2017.

[23] N. Sadoughi and C. Busso, "Retrieving target gestures toward speech driven animation with meaningful behaviors," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. ACM, 2015, pp. 115–122.

[24] H. X. Pham, Y. Wang, and V. Pavlovic, "End-to-end learning for 3d facial animation from raw waveforms of speech," *arXiv preprint arXiv:1710.00920*, 2017.

[25] H. X. Pham, S. Cheung, and V. Pavlovic, "Speech-driven 3d facial animation with implicit emotional awareness: a deep learning approach," in *The 1st DALCOM workshop, CVPR*, 2017.

[26] A. Fernández-Baena, R. Montaño, M. Antonijoan, A. Roversi, D. Miralles, and F. Alías, "Gesture synthesis adapted to speech emphasis," *Speech Communication*, vol. 57, pp. 331–350, 2014.

[27] C.-C. Chiu and S. Marsella, "How to train your avatar: A data driven approach to gesture generation," in *International Workshop on Intelligent Virtual Agents*. Springer, 2011, pp. 127–140.

[28] C. Pelachaud, N. I. Badler, and M. Steedman, "Generating facial expressions for speech," *Cognitive science*, vol. 20, no. 1, pp. 1–46, 1996.

[29] D. DeCarlo, M. Stone, C. Revilla, and J. J. Venditti, "Specifying and animating facial signals for discourse in embodied conversational agents," *Computer animation and virtual worlds*, vol. 15, no. 1, pp. 27–38, 2004.

[30] C. Busso, Z. Deng, M. Grimm, U. Neumann, and S. Narayanan, "Rigid head motion in expressive speech animation: Analysis and synthesis," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1075–1086, 2007.

[31] S. Levine, C. Theobalt, and V. Koltun, "Real-time prosody-driven synthesis of body language," in *ACM Transactions on Graphics (TOG)*, vol. 28, no. 5. ACM, 2009, p. 172.

[32] Y. Ding, C. Pelachaud, and T. Artieres, "Modeling multimodal behaviors from speech prosody," in *International Workshop on Intelligent Virtual Agents*. Springer, 2013, pp. 217–228.

[33] G. Hofer and H. Shimodaira, "Automatic head motion prediction from speech data," in *INTERSPEECH*, 2007.

[34] B. H. Le, X. Ma, and Z. Deng, "Live speech driven head-and-eye motion generators," *IEEE transactions on visualization and computer graphics*, vol. 18, no. 11, pp. 1902–1914, 2012.

[35] C. Ding, P. Zhu, L. Xie, D. Jiang, and Z.-H. Fu, "Speech-driven head motion synthesis using neural networks," in *15th Annual Conference of the International Speech Communication Association*, 2014.

[36] R. Lippa, "The nonverbal display and judgment of extraversion, masculinity, femininity, and gender diagnosticity: A lens model analysis," *J. Pers. Soc. Psychol.*, vol. 32, no. 1, pp. 80–107, 1998.

[37] A. Aly and A. Tapus, "A model for synthesizing a combined verbal and nonverbal behavior based on personality traits in human-robot interaction," in *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, March 2013, pp. 325–332.

[38] O. Celiktutan, E. Skordos, and H. Gunes, "Multimodal human-human-robot interactions (mhhri) dataset for studying personality and engagement," *IEEE Transactions on Affective Computing*, pp. 1–1, 2018.

[39] S. Petridis, T. Stafylakis, P. Ma, F. Cai, G. Tzimiropoulos, and M. Pantic, "End-to-end audiovisual speech recognition," *arXiv preprint arXiv:1802.06424*, 2018.

[40] J. Lyons, "Python speech features," https://github.com/jameslyons/python_speech_features, 2013, [Online; accessed 04/05/2018].

[41] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.

[42] C.-H. Chen and D. Ramanan, "3d human pose estimation= 2d pose estimation+ matching," in *CVPR*, vol. 2, no. 5, 2017, p. 6.

[43] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt, "Vnect: Real-time 3d human pose estimation with a single rgb camera," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, p. 44, 2017.

[44] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional 3d pose estimation from a single image," in *the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[45] Z. Zeng, Z. Zhang, B. Pianfetti, J. Tu, and T. S. Huang, "Audio-visual affect recognition in activation-evaluation space," in *2005 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2005, pp. 4–pp.

[46] P. Agarwal, S. Al Moubayed, A. Alspach, J. Kim, E. J. Carter, J. F. Lehman, and K. Yamane, "Imitating human movement with teleoperated robotic head," in *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2016, pp. 630–637.

[47] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Prentice Hall, 1999.

[48] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[49] E. Bozkurt, E. Erzin, and Y. Yemez, "Affect-expressive hand gestures synthesis and animation," in *2015 IEEE International*

Conference on Multimedia and Expo (ICME). IEEE, 2015, pp. 1–6.

[50] H. Pham, "Pyaudio: Portaudio v19 Python bindings," https://people.csail.mit.edu/hubert/pyaudio/, 2006, [Online; accessed 09/05/2018].

[51] J. Ondras, O. Celiktutan, E. Sariyanidi, and H. Gunes, "Automatic replication of teleoperator head movements and facial expressions on a humanoid robot," in *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2017, pp. 745–750.

[52] B. Rammstedt and O. P. John, "Measuring personality in one minute or less: A 10-item short version of the big five inventory in english and german," *Journal of research in Personality*, vol. 41, no. 1, pp. 203–212, 2007.

[53] "Softbank robotics," https://www.ald.softbankrobotics.com/en, [Online; accessed 27/04/2018].

[54] "Choregraphe simulation environment," http://doc.aldebaran.com/2-5/software/choregraphe/index.html, [Online; accessed 27/04/2018].

[55] "Pepper robot documentation," http://doc.aldebaran.com/2-4/family/pepper_technical/joints_pep.html, [Online; accessed 27/04/2018].

[56] B. Thompson, "Canonical correlation analysis," *Encyclopedia of statistics in behavioral science*, 2005.

[57] R. M. Krauss, P. Morrel-Samuels, and C. Colasante, "Do conversational hand gestures communicate?" *Journal of personality and social psychology*, vol. 61, no. 5, p. 743, 1991.

[58] P. Read and M.-P. Meyer, *Restoration of motion picture film*. Butterworth-Heinemann, 2000.

[59] T. Jolliffe and S. Baron-Cohen, "The strange stories test: A replication with high-functioning adults with autism or asperger syndrome," *Journal of autism and developmental disorders*, vol. 29, no. 5, pp. 395–406, 1999.

[60] "The MARY Text-to-Speech System," http://mary.dfki.de/, version 5.2. [Online; accessed 17/05/2018].

[61] M. McRorie, I. Sneddon, E. de Sevin, E. Bevacqua, and C. Pelachaud, "A model of personality and emotional traits," in *International Workshop on Intelligent Virtual Agents*. Springer, 2009, pp. 27–33.

[62] V. Fromkin, R. Rodman, and N. Hyams, *An introduction to language*. Cengage Learning, 2018.

[63] R. W. Roring, F. G. Hines, and N. Charness, "Age differences in identifying words in synthetic speech," *Human factors*, vol. 49, no. 1, pp. 25–31, 2007.

[64] S. J. Winters and D. B. Pisoni, "Perception and comprehension of synthetic speech," *Research on spoken language processing report*, no. 26, pp. 95–138, 2004.

Jan Ondras received BA and MEng degrees in Computer Science from the University of Cambridge in 2018. He then joined a machine learning startup Cognexa in Slovakia and later worked as a Research Assistant with the Institute for Creative Technologies, University of Southern California, USA. He is currently a Machine Learning Engineer working on video conferencing devices at Cisco in Norway. His research interests include the application of machine learning in the areas of human-robot interaction and affective computing.

Oya Celiktutan received the PhD degree in Electrical and Electronic Engineering from Bogazici University, Turkey, in collaboration with National Institute of Applied Sciences of Lyon, France, in 2013. She spent several years as a researcher at Queen Mary University of London, University of Cambridge and Imperial College London, respectively.

Since 2018, Dr. Celiktutan is an Assistant Professor (Lecturer) in the Centre for Robotics Research (CoRe) within the Department of Engineering, King's College London, UK, and she is the Head of Social AI & Robotics Laboratory. Her research focuses on computer vision and machine learning within the scope of human behaviour understanding, social signal processing, and human-robot interaction.

Paul Bremner received a BSc in Robotic and Electronic Systems Engineering from the University of Salford in 2003. He received an MSc in Advanced Technologies in Electronics and a PhD in Human-Robot interaction from the University of the West of England in 2005 and 2010 respectively. Since 2010 he has worked at the University of the West of England on a number of projects, first as a research associate then as a research fellow, where he is currently employed on the EPSRC-funded programme grant Robots for Nuclear Environments. His research interests include human-robot interaction, multi-modal communication, tele-presence, machine learning and robot ethics.

Hatice Gunes (SM'16) is a Reader / Associate Professor at the Department of Computer Science and Technology, University of Cambridge (UK) leading the Affective Intelligence and Robotics Lab. Her expertise is in the areas of affective computing and social signal processing cross-fertilising research in human behaviour understanding, computer vision, signal processing, machine learning, and human-robot interaction. She has published over 100 papers in these areas. Dr Gunes is the President-Emeritus (2017-2019) of the Association for the Advancement of Affective Computing, and was the General Co-Chair of ACII 2019, and the Program Co-Chair of ACM/IEEE HRI 2020 and IEEE FG 2017. Her research has been supported by various competitive grants, with funding from EPSRC, Innovate UK, British Council and EU Horizon 2020. She is a Fellow of the Engineering and Physical Sciences Research Council UK (EPSRC), a Faculty Fellow of the Alan Turing Institute, and a Staff Fellow of Trinity Hall Cambridge.