Audio-driven Laughter Behavior Controller

Yu Ding[*], Jing Huang[†] and Catherine Pelachaud[‡][§]

August 26, 2022

Abstract

It has been well documented that laughter is an important communicative and expressive signal in face-to-face conversations. Our work aims at building a laughter behavior controller for a virtual character which is able to generate upper body animations from laughter audio given as input. This controller relies on the tight correlations between laughter audio and body behaviors. A unified continuous-state statistical framework, inspired by Kalman filter, is proposed to learn the correlations between laughter audio and head/torso behavior from a recorded laughter human dataset. Due to the lack of shoulder behavior data in the recorded human dataset, a rule-based method is defined to model the correlation between laughter audio and shoulder behavior. In the synthesis step, these characterized correlations are rendered in the animation of a virtual character. To validate our controller, a subjective evaluation is conducted where participants viewed the videos of a laughing virtual character. It compares the animations of a virtual character using our controller and a state of the art method. The evaluation results show that the laughter animations computed with our controller are perceived as more natural, expressing amusement more freely and appearing more authentic than with the state of the art method.

keywords: laughter, audio-driven, data-driven, animation synthesis, continuous-state, Kalman filter, prosody, nonverbal behaviors, virtual character, statistical framework.

---

[*] Y. Ding is at University of Houston, Houston, Texas, USA.

[†] J. Huang is at Zhejiang Gongshang Univeristy, School of Information and Electronic Engineering, Hangzhou, China.

[‡] C. Pelachaud is at ISIR-CNRS, Universit Pierre et Marie Curie, Paris, France.

[§] Work conducted while the authors were at LTCI-CRNS, Télécom-ParisTech.

1 Introduction

Laughter is universal and prevalent throughout all known cultures [1]. It has been observed that newborns that are only a few months old have the ability to laugh even though they are not able to speak [2]. Deaf-blind children laugh in a manner that is fundamentally similar to the way in which normal-hearing individuals laugh [3].

Laughter is an important communicative and expressive signal [3]. It regulates speech during a conversation [4] and indicates social group belonging [5]. It is often used during the formation and maintenance of social groups to smoothly and positively manage relationships [6]. For instance, it can be viewed as a positive feedback when interlocutors laugh at the same time. Also, it is also employed to elicit interlocutors' laughter [4].

Laughter is a remarkable indicator of interlocutors' emotions during conversations. Especially, laughter occurs frequently to convey positive emotions or cheerful mood in human conversations [7]. For example, humans laugh at humorous stimuli or to express their pleasure when receiving praise statements [8]. Additionally, laughter may be related to other emotional states in human communication. For instance, humans often laugh when feeling embarrassed, disappointed, stressed or even cynical [1] [9]. Huber and Ruch [9] distinguish up to 23 different types of laughter ranging from hilarious, hysterical to embarrassed, desperate, contemptuous.

Laughter is a multimodal process involving facial expressions, body movements and vocalizations [3]. For hilarious laughter, muscular activities include mainly the zygomatic major, mouth opening and jaw movement. Orbicularis oculi muscle is squinted as for Duchenne smile [10]. Eyebrows may be raised or even frown in very intense laughter [3]. Saccadic movements affect the whole body. Torso may bend back and forth and shoulder may shake. Changes in respiration patterns are also prominent. Inhalation and exhalation phases are very noticeable. All these movements are done very rhythmically [3]. They are also highly correlated and they arise from the same physiological processes [3]. Furthermore, several researchers proposed to utilize laughter body movements to distinguish hilarious laughter from social laughter [11] [12] and laughter from other communicative activities (e.g. non-laughter) [13] [14].

Recently researchers have proposed models for laughter generation, detection, and perception. For instance, a game scenario was set up to induce natural hilarious laughter interaction between human participants and a virtual character [15] [16]; McKeown et al. [17] looked into the effect of environmental and social factors, such as a friendly, relaxing environment by playing popular music and a strong social relationship (i.e. friends), on inducing hilarious laughter and conversational laughter.

In recent years, virtual characters have become increasingly popular in several applications of human-computer interactions, such as social coaching [18], companionship [19] [20] and museum guide [21]. They have been endowed with human-like emotional, social and communicative qualities [22] [23] [24] [25] [26] [27] [28]. Lately, particular efforts have been made to add laughter as such a quality to the virtual character [29] [30] [31]. Our work lies in this research direction. We aim to develop an expressive virtual character able to interact naturally with humans. One step toward this direction is to allow the virtual character to display a large palette of socio-emotional behaviors. In this paper, we focus on a particular behavior, namely hilarious laughter.

Our aim is to build a laughter behavior controller able to generate laughter the animations of the upper body (head, torso, and shoulders) from laughter audio given as input. The laughter behavior controller consists of a unified statistical framework for head and torso animations and a rule-based framework for shoulder animation. Our underlying idea is based on the tight correlation between laughter audio and laughter motions. While a statistical framework is developed to capture the correlation between laughter audio and head/torso motion from a recorded laughter human dataset, a rule-based framework is proposed, due to the lack of shoulder behavior data in the recorded human dataset, to define the correlation between laughter audio and shoulder motion. Once these correlations are captured or defined, laughter animations can be calculated from laughter audio in the synthesis step. (Synthesizing laughter audio are beyond our research topic.)

In Section 2, we review previous works on laughter animation generation. Then, Section 3 presents a motion capture dataset of human laughter used to train the proposed statistical model. Next, Section 4 introduces our laughter behavior controller, including the statistical framework and the rule-based framework. Later on, Section 5 describes a subjective evaluation we designed to validate our laughter behavior controller. Section 6 reports the evaluation results and Section 7 discusses the results. Finally, Section 8 concludes this work and summarizes its contributions.

2 Related Work

Recently, several works have been dedicated to simulating laughter. Models have been proposed to compute laughter lip animation [32] [33] [34], laughter facial expression [35] [36] [29], laughter head motion [29], laughter torso shaking [37] [29] [38], and laughter shoulder trembles [39]. In these works, laughter audio is used as input signals to compute the output appropriate laughter animations. This section reviews briefly these works.

DiLorenzo et al. [37] proposed a physics-based laughter torso model. This model with manually-defined parameters takes the air flow of laughter audio as input to infer torso deformation configured by spine and clavicle motions and respiratory muscles. A force parameter is inferred from the air flow and used to animate respiratory muscles, spine, and clavicle. Such an approach could not be generalized to other motion modalities (e.g. head rotation animation) being independent of the air flow.

Niewiadomski et al. [39] conducted a spectrum analysis on laughter shoulder movements and characterized the relationship between laughter audio and shoulder movements. The harmonic signals are used to produce real-time trembling shoulder animations from the input laughter audio.

Niewiadomski and Pelachaud [36] found that laughter facial expression is related to the intensity of laughter audio. They indicated that laughter intensity can be used to infer facial motion but they do not report the inference procedure. Later on, Niewiadomski et al. [40] proposed another approach to infer facial expression. Their approach is based on selecting whole motion episodes from a motion capture dataset of laughter episodes. The selection process takes into account two factors: the intensity and the duration of laughter episode. In their work, the intensity of each episode is assumed to be constant and audio prosody is ignored. As such this work lacks to capture the synchronization mechanism between laughter motion and laughter audio.

Similar to Niewiadomski et al. [40], Urbain et al. [41] proposed to replicate motion by selecting facial expression of human laughter episode from a motion capture dataset. The selection is done based on only two variables: mean and standard deviation between the recorded audio and the input audio. This may not be enough to characterize long audio sequence.

Cosker and Edge [35] used Hidden Markov Models (HMMs) to synthesize laughter facial motion from audio features. The authors built subject-specific HMMs to model laughter audio and motion. To compute the laughter animation of a new subject, the first step is to select one HMM from the set of HMMs by comparing the audio similarity between the new subject and the subjects involved in the training dataset. Then the selected HMM is used to produce the output laughter animation from the most likely state sequence. However, if one state in the state sequence may last very long, it would lead to still motion, which would produce unnatural animation.

Çakmak et al. [32] [42] decomposed the sequences of facial expression of laughter into 3 segments: *Neutral* (N), *Smile* (S) and *Audible Laugh* (L). Three motion segment subsets are collected from a human dataset. Each subset is used to train an HMM. In the synthesis step, a label sequence ($[N, L, N]$ or $[S, L, S]$) and the lasting time of each label are used as input. The lasting time information ensures that facial expression labeled by L begins and ends at the same time with laughter sound. While this approach could be validated for short laughter audio, it could lead to unnatural animations in long episodes as a state in HMM representing a position could last a long time. Additionally, laughter animation intensity is assumed to be uniform within and across episodes. However, laughter intensity can be largely different between episodes and can temporally vary during laughter sound.

Ding and colleagues proposed different models aiming at simulating the motion qualities of laughter.

Ding et al. [29] have attempted to generate facial expressions (mouth region and upper face) and behaviors of head and upper torso from laughter audio signals. Linear regression method is applied to predict lip and jaw animations, where laughter pseudo-phonemes (defined by Urbain et al. [41] in reference to speech phonemes) are used to estimate mouth shape, and, audio prosody features are used to configure the openness amplitude of the mouth shape. Similar to Niewiadomski et al. [40] and Urbain et al. [41], selecting motion samples from human data is used to produce head and upper face animations. A manually-defined Proportional Derivative (PD) controller is proposed to compute torso behaviors. The definition of the PD controller relies on the assumption that head motion follows torso movements during laughter.

In a successive work, Ding et al. [38] proposed a statistical framework combining Coupled HMM and Parametric HMM to synthesize head and torso behaviors using the input laughter audio signals. Coupled HMM allows capturing the temporal relationship between head and torso behaviors; Parametric HMM acts as obtaining the closed correlation between audio signals and behaviors.

Ding et al. [33] focused on computing laughter lip animations. The underlying idea is to infer mouth shape from the laughter pseudo-phonemes and prosody features, based on GMM. Then an HMM-based interpolation function is trained on human data. This specific interpolation function is capable of capturing the subtle co-articulation of the lip motions. In the synthesis step, the built GMM and HMM are both used to configure the lip shape at each frame.

In the above works, HMM has been applied to generate laughter animation. In fact, HMM is a set of discrete states and fits very well to model discrete variables (e.g. speech phonemes or words). However, when modeling laughter animation, continuous behavior trajectory is regarded as a sequence of discrete positions in [35] [32] [38] [42]. It means that the information of the dynamics and of the continuity in behavior trajectory could be lost and could not be captured by HMM. Tokuda et al. [43] and Brand [44] propose HMM-based synthesis algorithms to generate continuous speech signals and facial expressions respectively. Their algorithms make the assumption that state transition probabilities are time-invariant. Such probabilities reflect the overall bias towards all the training data. They could be untrue in each testing sequence and could cause over-smoothing signal trajectories when a state lasts for a long time. Considering that laughter movements often tremble, the algorithms from [43] [44] could not be suitable to compute laughter animations.

In our work, we develop a continuous-state statistical framework to generate laughter behavior. This framework has not only the advantage of modeling the data sequence as HMM does but it also avoids segmenting continuous signals into discrete variables.

3 Human Laughter Dataset

As mentioned before, a statistical framework is proposed to build a laughter behavioral controller. Its underlying idea is to automatically capture the implicit data relationship between laughter behavior and laughter audio from human motion dataset. To reach this goal, a human laughter dataset is recorded; then it is used to train the statistical framework.

The motion capture sessions took place in an anechoic chamber. 8 participants (6 males and 2 females) were recruited. During the recording session, the participant sat in front of a PC and watched funny movies for about 25-40 minutes. The funny movies had been attentively selected for eliciting spontaneous laughter in participants. Figure 1 shows the front and the side views of a laughing participant who is watching funny movies. The data collection involves one participant at a time.

The laughter sound was recorded by a headset microphone at 44100 Hz (see Figure 1). The participant was equipped with the motion capture system Xsens, which consists of a headband, two sensors on the right and the left shoulders and three sensors on the torso (placed at upper, middle and lower positions along the torso). The headband samples the 3-dimensional head rotation angles at 125 frames per second (fps). Unfortunately, the two

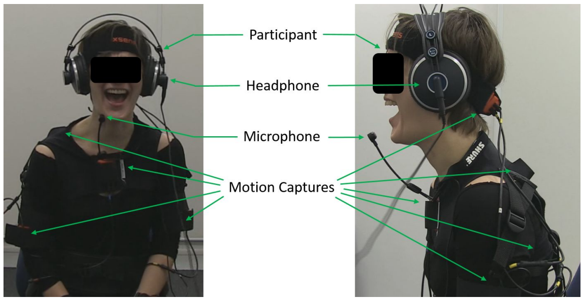


Figure 1: Left figure: front view. Right figure: side view. The participant is laughing when watching funny movies. She is equipped with a motion capture system and a microphone.

sensors on the shoulders did not capture shoulder movements successfully. Moreover, although three sensors are used to track the torso movement, the rotation angles captured by the three sensors are linearly related to each other. It means that only the 3 rotation angles from one sensor are significant; the 6 rotation angles from the other two sensors can be linearly interpreted from the first sensor. The torso movement was sampled at 125 fps. The impossibility to capture the more detailed movement of the shoulder and of the torso is due to the limitation of the motion capture system we used.

During data processing, all laughter episodes were manually picked up by excluding moments with only speech or silence. Finally, a total of 505 laughter episodes were collected. Each one lasts between 1 and 37 seconds. It contains the head and torso movement data and the audio signal.

Laughter movements involve many signals. Human laughter, while sharing many properties and qualities, shows high individual differences in its acoustic and visual production. Capturing and modeling the individual differences of laughter is beyond the scope of our current work. To overcome this issue, our work uses the audiovisual data of one participant who has been subjectively selected as the most spontaneous one among all the participants. We obtained a dataset of 78 episodes, each one lasting between 2 and 34 seconds.

4 Laughter Behavior Controller

Our controller embeds two modules, a unified data-driven approach for head and torso motions and a rule-based approach for the shoulder motion. The data-driven approach (see Section 4.2) is proposed to extract the temporal relationship between human laughter audio and the accompanied head/torso motion. Since shoulder movements are unavailable in the laughter data collection, a rule-based approach is designed to define the relationship between laughter audio and shoulder movements (see Section 4.3). In the synthesis step, the extracted and defined relationships act as a function taking laughter audio as input and outputting human-like head, torso and shoulder animations.

4.1 Audio and Visual Features

We use the Greta virtual character [45] to visualize the laughter animation. Its skeleton follows the MPEG-4 standard [46]. We control the upper body with 3 joints on the neck and head, 2 on the right and the left shoulders, and 8 on the spine. Each joint is animated by 3 rotation angles. The head, the torso and the shoulders of the virtual character can be animated by configuring neck, spine and shoulder joints.

Head and torso features. Since the adjacent joints on the neck/spine are closely related to each other, a joint moves always by following its adjacent skeletal joints. To overcome the lack of data for all human spine and neck joints, the relationships between adjacent joints are modeled by Proportional-Derivative (PD) controllers. Such an approach has also been employed in [37]. The parameters of the PD controllers are configured by hand as in [37]. In our work, the top joint of the neck (respectively spine) joints (defined by the skeletal hierarchy) is selected as the head (resp. torso) joint that leads the PD controller. So, if the head (respectively torso) joint is known, the movements of the other joints on the neck (resp. spine) can be inferred using the PD controllers. Hence, 6 rotation angles are taken as head and torso motion features: 3 for animating the head and 3 for animating the torso. During the synthesis phase, the values of the 6 motion features are inferred at each time step.

Shoulder feature. Shoulder motions are viewed as the combination of the up-down and the forward-backward movements. In our work, the up-down and the forward-backward movements are considered to be linearly related to each other. Moreover, the right and the left shoulders are assumed to follow symmetric motions. Hence, only one dimension feature is used to animate the right and the left shoulders.

Audio feature. The speech processing software Praat [47] is used to extract the audio feature loudness from the recorded laughter audio at 125 fps, which matches with the sample frequency of the motion capture data, 125 *fps*.

4.2 Head and Torso Behavior Controller

The data-driven approach is proposed to build head and torso behavior controller. It is capable of computing 6 motion features at each time step according to the laughter audio input. In particular, a unified statistical framework is proposed to generate the trajectories of the 6 motion features (3 head and 3 torso motion features). For the sake of simplicity, the 6 motion features are not distinguished in the following framework description. A generic motion feature sequence is noted by m, as follows:

$$m = [m_1, ..., m_t, ..., m_T] \tag{1}$$

where m_t is the value of m at time t. m stands for any one of the 6 motion features sequences. Since human motion is always continuous without breaks, m_t is a continuous variable. It can be viewed as a sample from a continuous space.

One classical approach to model continuous variables is to quantify them as discrete variables, also called state variables in the previous works [38] [33] [35] [32] [42]. It is referred to as discrete-state framework in this work. The underlying idea is to represent continuous variables with a set of discrete variables. Particularly, a continuous variable is approximated by a discrete value or a weighted sum of a set of discrete variables. While such an approximation may be able to model and capture some temporal relationship of sequential data, it may also lead to some loss of information at each time frame. And thus, it may result in degrading data accuracy, continuity and dynamics.

To avoid the shortcomings arising from data discretization, a continuous-state framework is proposed to model the motion temporal relationship. In the continuous-state framework, the state variables are viewed as samples from a continuous space instead of a discrete space (a set of discrete variables).

To build the continuous space, m_t and its velocity feature (the first order derivative), Δm_t, is combined as a joint vector as follows:

$$o_t = [m_t, \Delta m_t]^\top \tag{2}$$

where $\top$ stands for the transpose of a vector. Since m_t and Δm_t are continuous variables, o_t is a 2-dimensional continuous variable and a sample from a 2-dimensional continuous space, denoted by $\mathbb{S}$. Furthermore, the probability distribution of o_t in $\mathbb{S}$ is modeled by a Gaussian probability distribution,

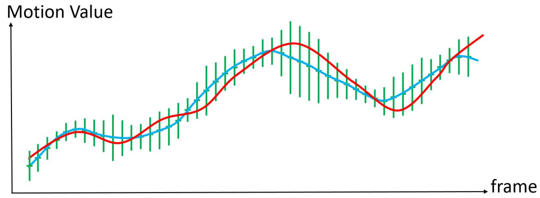


Figure 2: Illustration of continuous state. The blue curve represents the mean values of Gaussian probability distribution (*gpd* or s_t) at each time step. The width of green lines represents the variance of *gpd* at each time step. The red curve represents a sample from the *gpd* at each time step. As can be seen, the mean and the variance are both continuous variables in time. This is why the sampling sequence (red curve) is smooth.

denoted by *gpd* and defined by a mean vector and a covariance matrix. The mean vector indicates a sample which occurs with the highest probability; the covariance matrix is a measure of the sample dispersion around the mean vector. The mean vector and the covariance matrix are not uniform for o_t at different time steps. It means that *gpd* varies along with the time step, t. Such a *gpd* is named motion state, noted by s_t at time t.

Since the mean vector and the covariance matrix are continuous variables, s_t is a continuous state. Figure 2 illustrates an example of motion state, where the motion velocity feature is ignored and only the motion feature is depicted. As can be observed, the mean and the variance of *gpd* are both continuous variables; they vary smoothly along with the time step, t, which is helpful to produce a smooth curve without any discontinuities. No additional specific smoothing operations or interpolation techniques are employed to smooth the produced animation trajectory.

The behavior trajectory value and its velocity at each time step are available from human data but their probability distributions are unknown. So, while o_t is observable, s_t is unobserved from human data. It is a hidden variable.

s_t is characterized by a *gpd* at time step t; and o_t is a sample from s_t. That is o_t follows s_t. The underlying idea of synthesizing animations is to first determine s_t from the input audio signals and then to estimate o_t (or m_t) from s_t. The synthesis process is summarized by the two following steps:

1. Step 1: *Motion State Determination*. The motion state, s_t, is determined from the audio feature a_t and s_{t-1}. This step is described in Section 4.2.1.

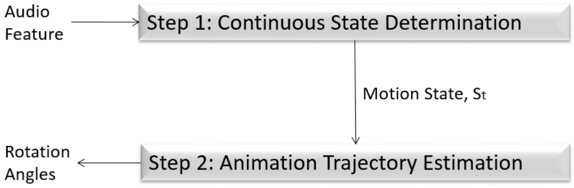

Audio Feature
Step 1: Continuous State Determination
Motion State, St
Rotation Angles
Step 2: Animation Trajectory Estimation

Figure 3: Synthesis overview. Step 1 (see Section 4.2.1) determines the Gaussian probability distribution at each time step, s_t, according to the input audio signal, a_t; Step 2 (see Section 4.2.2) infers the motion feature, m_t, from s_t.

2. Step 2: *Motion Trajectory Estimation*. A laughter animation stream is estimated from s_t. This step is introduced in Section 4.2.2.

Figure 3 illustrates the overview of synthesizing head and torso animations. s_t can be viewed as a mediate or hidden variable. It is the output of the first step and the input of the second step.

4.2.1 Step 1: Motion State Determination

The first step is to determine s_t from the audio feature a_t. In our framework, s_t is assumed to explicitly depend on s_{t-1} and a_t. It is conditionally independent of the audio features and the motion states at the earlier time steps. Such dependency relationships are illustrated in Figure 4. Based on such a framework, the key question in Step 1 is to solve s_t from s_{t-1} and a_t. Then the solved s_t and the input audio at next time, a_{t+1}, are used as input to solve s_{t+1}. Such a solution process is carried out along with the time step, t, as can be seen in Figure 4.

The underlying idea is first to infer two estimates of s_t respectively from s_{t-1} and from a_t. The estimate from s_{t-1} is denoted by s_t^s while the other from a_t by s_t^a; secondly the idea is to fuse these two estimates (s_t^s and s_t^a) as the final estimate of s_t (the output of Step 1). So, s_t^s, s_t^a and s_t are defined as follows:

- s_t^a: the estimate of s_t depending on a_t.
- s_t^s: the estimate of s_t depending on s_{t-1}.
- s_t: the estimate of s_t depending on both s_{t-1} and a_t.

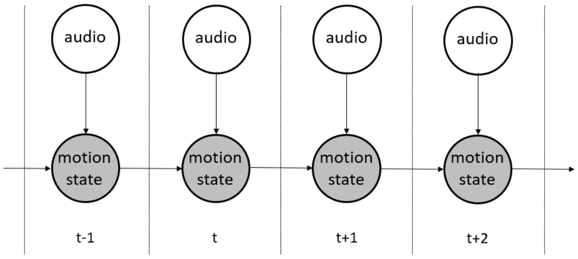


Figure 4: Dependence relationship between motion state (s_t) and audio feature (a_t). The motion state at time t, denoted as s_t, depends on the motion state at time $t-1$, denoted as s_{t-1}, and the audio feature at time t, denoted as a_t; it is conditionally independent of $s_{t'-1}$ and $a_{t'}$ (where $t' < t$) at the earlier times. In the synthesis step, s_t is determined from a_t and s_{t-1}. This determination process is illustrated in Figure 5.

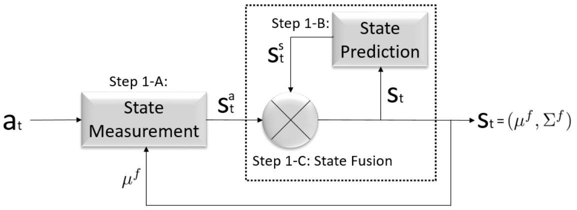


Figure 5: Overview of motion state determination (Step 1). Step 1-A is conducted by the developed partial parametric K-means algorithm; Step 1-B and Step 1-C framed in dash are conducted by Kalman filter.

More precisely, s_t^s, s_t^a and s_t are defined respectively by a *gpd*. Step 1 is to estimate respectively two *gpd*s (s_t^s and s_t^a) and then to fuse these two *gpd*s as a new *gpd* defining s_t.

Figure 5 illustrates the determination of s_t from s_{t-1} and a_t, which can be summarized in three sub-steps:

1. *Step 1-A: from a_t to s_t^a.* Step 1-A estimates the role of a_t (audio feature at the current time) on s_t (motion state at the current time). The estimated result is s_t^a. Step 1-A is represented by the operation called *State Measurement* in Figure 5.

2. *Step 1-B: from s_{t-1} to s_t^s.* Step 1-B estimates the role of s_{t-1} (motion state at previous time t-1) on s_t (motion state at the current time). The estimated result is s_t^s. Step 1-B is represented by the operation

called *State Prediction* in Figure 5.

3. *Step 1-C: from s_t^s and s_t^a to s_t.* Step 1-C is to fuse the estimates obtained at Step 1-A and at Step 1-B. The fused result is s_t. Step 1-C is represented by the operation called *State Fusion* in Figure 5.

Step 1-A and Step 1-B are independent of each other; they can be applied in any order. In fact, Step 1-A, Step 1-B and Step 1-C are conducted once at each time step. The output s_t from Step 1-C at time step t is used as input to Steps 1-A and 1-B at time step $t+1$, which can be observed in Figure 5. In particular, the mean vector and the covariance matrix of s_t are both used as input to Step 1-B, while only the mean vector of s_t is used as input to Step 1-A.

Step 1-A is carried out by a new statistical framework, called a partial parametric K-means algorithm. Step 1-B and Step 1-C relies on Kalman filter. (The description of Kalman filter is beyond the scope of this paper.) Step 1-A, Step 1-B, and Step 1-C will be introduced as follows.

Step 1-A: Audio Feature a_t to Motion State s_t^a

To compute s_t^a, a partial parametric k-means clustering is developed. It is a standard k-means clustering where the partial elements of the mean vectors are conditional on the contextual variable(s) and the other elements are fixed.

Carrying out a partial parametric k-means clustering consists of two steps. The first step splits the data into clusters by conducting a standard k-means clustering. Each cluster is represented by a mean vector and a covariance matrix. The second step learns the relationship between the partial elements of the mean vectors and the contextual variable(s). These two steps are detailed as follows.

A partial parametric k-means clustering is carried out on the training set, denoted by $\{sp_t^k\}$, where *sp* is the abbreviation of sample. sp_t^k stands for a training sample at time t from the k-th training sequence. sp_t^k is a 5-dimensional joint vector defined as follows,

$$sp_t^k = [m_{t-1}^k, \Delta m_{t-1}^k, m_t^k, \Delta m_t^k, a_t^k]^\top$$

where sp_t^k is comprised of the audio feature at the current time step and of the motion features and their derivative features at the current and the previous time steps. The standard k-means clustering is conducted to split $\{sp_t^k\}$ into J groups. The probability distribution of the samples from each group are modeled by a *gpd* with a diagonal covariance matrix. The mean vector of the *gpd* is a 5-dimensional vector. The j-th *gpd* is characterized by its mean vector, μ_j, and the covariance matrix, Σ_j, as follows:

$$\mu_j = [\mu_j^{m^{-1}}, \mu_j^{\Delta m^{-1}}, \mu_j^{m^0}, \mu_j^{\Delta m^0}, \mu_j^{a^0}] \tag{3}$$

$$\Sigma_j = diag[\Sigma_j^{m^{-1}}, \Sigma_j^{\Delta m^{-1}}, \Sigma_j^{m^0}, \Sigma_j^{\Delta m^0}, \Sigma_j^{a^0}] \tag{4}$$

where -1 and 0 at the top-right corner respectively stand for the previous and the current time steps. For example, $\mu^{m^{-1}}$ and $\mu^{\Delta m^{-1}}$ respectively represent the means of m_{t-1}^k and of Δm_{t-1}^k at the previous time step.

The partial elements of the mean vector, μ_j, is assumed to be related to the contextual variable. In our work, these elements include μ^{m^0} and $\mu_j^{\Delta m^0}$; and the contextual variable is the audio feature at the current time step, a_t. In particular, $\mu_j^{m^0}$ and $\mu_j^{\Delta m^0}$ is picked up from μ_j, noted by $\mu_j^a = [\mu_j^{m^0}, \mu_j^{\Delta m^0}]$, named parametric mean vector. For the sake of simplicity, μ_j^a is assumed to be linearly conditional on a_t. The dependence is formulated as follows:

$$\mu_j^a(a_t) = W^j a_t + \bar{\mu}^j \tag{5}$$

where W^j is a 2×1 matrix; and $\bar{\mu}^j$ is an offset vector. W^j and $\bar{\mu}^j$ are learned through Least Mean Square algorithm (LMS).

According to the description above, the partial parametric k-means clustering can be summarized as follows.

$$\mu_j = [\mu_j^{m^{-1}}, \mu_j^{\Delta m^{-1}}, \mu_j^a, \mu_j^{a^0}] \tag{6}$$

$$\mu_j^a = [\mu^{m^0}, \mu^{\Delta m^0}] \tag{7}$$

$$\mu_j^a(a_t) = W^j a_t + \bar{\mu}^j \tag{8}$$

$$\Sigma_j = diag[\Sigma_j^{m^{-1}}, \Sigma_j^{\Delta m^{-1}}, \Sigma_j^{m^0}, \Sigma_j^{\Delta m^0}, \Sigma_j^{a^0}] \tag{9}$$

which describes the j-th *gpd* in the partial parametric k-means clustering. This partial parametric k-means clustering is denoted by Γ^0. Then two new k-means clustering are extracted from Γ^0 by splitting the elements in μ_j and in Σ_j into two parts.

The first one is obtained by ignoring μ_j^a in μ_j and the corresponding elements ($\Sigma_j^{m^0}$ and $\Sigma_j^{\Delta m^0}$) in Σ_j and keeping the other elements in μ_j and Σ_j, as follows:

$$\mu_j^s = [\mu_j^{m^{-1}}, \mu_j^{\Delta m^{-1}}, \mu_j^{a^0}] \tag{10}$$

$$\Sigma_j^s = diag[\Sigma_j^{m^{-1}}, \Sigma_j^{\Delta m^{-1}}, \Sigma_j^{a^0}] \tag{11}$$

This is named Γ^s (s is the abbreviation of standard), which is a standard k-clustering. The elements in the mean vector are fixed, once Γ^s is built.

The second one is obtained by only keeping μ_j^a in μ_j and the corresponding elements ($\Sigma_j^{m^0}$ and $\Sigma_j^{\Delta m^0}$) in Σ_j and ignoring the other elements in μ_j and Σ_j, as follows:

$$\mu_j^a = [\mu_j^{m^0}, \mu_j^{\Delta m^0}] \tag{12}$$

$$\mu_j^a(a_t) = W^j a_t + \bar{\mu}^j \tag{13}$$

$$\Sigma_j^a = diag[\Sigma_j^{m^0}, \Sigma_j^{\Delta m^0}] \tag{14}$$

This k-means clustering is named Γ^a (a is the abbreviation of audio), where all the elements ($\mu_j^{m^0}$, $\mu_j^{\Delta m^0}$) in the mean vector depend on a_t.

In Step 1-A, the input signals contain $[\mu^{m^{-1}}, \mu^{\Delta m^{-1}}]$ and a_t. The output signals contain $\{\mu_j^{m^0}, \mu_j^{\Delta m^0}\}$ and $\{\Sigma_j^{m^0}, \Sigma_j^{\Delta m^0}\}$ which describe s_t^a.

As can be observed, Γ^s and Γ^a respectively characterize the probability distributions of the input signals ($\mu^{m^{-1}}$, $\mu^{\Delta m^{-1}}$ and a_t) and the output signals (motion and velocity at time step t).

Step 1-A is carried out by selecting a cluster and then calculating s_t^a, as follows.

1. **Selecting Cluster**. The input signals (a_t, $\mu^{m^{-1}}$ and $\mu^{\Delta m^{-1}}$) are applied to Γ^s by computing the posterior probability of each cluster in Γ^s. The cluster probability with the highest posterior probability is selected. Its index is noted by j'.

2. **Calculating** s_t^a. j' is applied to Γ^a. $\Sigma_{j'}^a$ is extracted from the j'-th cluster in Γ^a. The input a_t is used to calculate $\mu_{j'}^a$ using Equation 13. $\Sigma_{j'}^a$ and $\mu_{j'}^a$ are taken as the output to define s_t^a.

We observe that a_t is used not only to select the cluster but also to calculate the parametric mean vector of the selected cluster. Additionally, the step of Selecting Cluster takes not only a_t but also the mean vector of s_{t-1} as input.

Step 1-B: Motion State s_{t-1} to Motion State s_t^s

This sub-step computes s_t^s, which is the estimate of s_t by taking into account only s_{t-1}. s_t^s will be fused into s_t with s_t^a in Step 1-C.

In Step 1-B, s_{t-1} (the input signal) and s_t^s (the output signal) respectively model the probability distributions of o_{t-1} and o_t. They are respectively characterized by a pair of $\{\mu_{t-1}, \Sigma_{t-1}\}$ and a pair of $\{\mu_t^s, \Sigma_t^s\}$, where $\mu_{t-1} = [\mu_{t-1}^{m}, \mu_{t-1}^{\Delta m}]^{\top}$, $\Sigma_{t-1} = diag[\Sigma_{t-1}^{m}, \Sigma_{t-1}^{\Delta m}]$, $\mu_t^s = [\mu_t^m, \mu_t^{\Delta m}]^{\top}$ and $\Sigma_t^s = diag[\Sigma_t^m, \Sigma_t^{\Delta m}]$. Therefore, Step 1-B is formulated as inferring $\{\mu_t^s, \Sigma_t^s\}$ according to $\{\mu_{t-1}, \Sigma_{t-1}\}$.

Since $\mu_{t'}^{m}$ and $\mu_{t'}^{\Delta m}$ (t'=t or $t-1$) respectively stand for the means of motion feature and of velocity feature, one can derive μ_t^s from μ_{t-1} by calculating:

$$\mu_t^s = F\mu_{t-1} \tag{15}$$

where

$$F = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \tag{16}$$

where motion velocity is assumed to be invariable at time steps t and $t-1$. Furthermore, Kalman filter provides the solution to the covariance matrix, Σ_t^s as follows.

$$\Sigma_t^s \quad = \quad F\Sigma_{t-1}F^{\top} \tag{17}$$

According to Equation 15 and Equation 17, the pair of $\{\mu_t^s, \Sigma_t^s\}$ is derived from the pair of $\{\mu_{t-1}, \Sigma_{t-1}\}$. That is s_t^s is inferred from s_{t-1}.

Step 1-C: Fusion of Motion States, s_t^s and s_t^a

Step 1-A and Step 1-B have derived the two estimates of s_t: s_t^a and s_t^s. Step 1-C consists in fusing s_t^a and s_t^s. The fused result is viewed as the output of Step 1, s_t.

s_t^a and s_t^s characterize the probability distribution of o_t with two different *gpd*s, which is respectively characterized by (μ^a, Σ^a) and by (μ^s, Σ^s). s_t is described by (μ^f, Σ^f). Step 1-C is to infer (μ^f, Σ^f) from (μ^a, Σ^a) and (μ^s, Σ^s). μ^f is a 2-dimensional vector and consists of μ_f^m and $\mu_f^{\Delta m}$. Σ^f is a diagonal matrix of $diag[\Sigma_f^m, \Sigma_f^{\Delta m}]$.

In our work, $\mu_f^{\Delta m}$ and $\Sigma_f^{\Delta m}$ are respectively estimated by $\mu^{\Delta m^0}$ in μ^a and $\Sigma^{\Delta m^0}$ in Σ^a, as follows.

$$\mu_f^{\Delta m} \quad = \mu^{\Delta m^0} \tag{18}$$

$$\Sigma_f^{\Delta m} \quad = \Sigma^{\Delta m^0} \tag{19}$$

μ_f^m and Σ_f^m is solved by fusing two *gpd*s: $(\mu^{m^0}, \Sigma^{m^0})$ and (μ^m, Σ^m), which are respectively extracted from (μ^a, Σ^a) and (μ^s, Σ^s). Kalman filter provides the solution to μ_f^m and Σ_f^m as follows:

$$\mu_f^m = \frac{\mu_a \Sigma^{m^0} + \mu_s \Sigma^m}{\Sigma^{m^0} + \Sigma^m} \tag{20}$$

$$\Sigma_f^m = \frac{\Sigma^{m^0} \Sigma^m}{\Sigma^{m^0} + \Sigma^m} \tag{21}$$

Equations 18-21 provide the solutions of $\mu_f(=[\mu_f^m, \mu_f^{\Delta m}])$ and $\sigma_f(=diag[\Sigma_f^m, \Sigma_f^{\Delta m}])$ according to s_t^a and s_t^s. μ_f and σ_f characterize the output of Step 1, s_t.

4.2.2 Step 2: Generating Motion Trajectory

Generating motion trajectory consists in synthesizing the output animation sequence from the motion state which has been inferred at each time step at Step 1. Each motion state is characterized by a *gpd*. It describes the probability distribution of the joint vector consisting of motion feature and its velocity.

A simple method can be used to output animation sequence. It is to concatenate the motion mean vector ($\mu_f = [\mu^m, \mu^{\Delta m}]$) of s_t at each time step and then to apply a specific interpolation technique (e.g. spherical cubic interpolation [48]) to smooth the concatenated sequence of motion mean vector. However, such a method ignores the relationship between the motion feature and the velocity feature as the two features are viewed as being independent of each other. Additionally, the specific interpolation technique is unrelated to the real data in a recorded mocap dataset. Such a method could degrade the naturalness and the dynamics of animations, although it could ensure their smoothness. It means that a pertinent interpolation technique should ensure both the smoothness and the dynamics of animations. To address this problem, we take into account the relationship between the motion feature and the velocity feature in the step of generating motion trajectory.

The data stream generation is derived by maximizing $P(m|s)$, with respect to m, where s is a sequence of s_t, noted as $s = [..., s_{t-1}, s_t, s_{t+1}, ...]$ and m is a sequence of m_t, noted as $m = [..., m_{t-1}, m_t, m_{t+1}, ...]^\top$. The solution to m is inspired by Tokuda et al. [43]. To solve m we use:

$$o = Wm \tag{22}$$

where o is a vector concatenating o_t at each time step, denoted as $o = [..., o_{t-1}^\top, o_t^\top, o_{t+1}^\top, ...]^\top$, where $o_t = [m_t, \Delta m_t]^\top$ (see Equation 2); and W is a $(2T) \times (T)$ operation matrix (T is the time length of o and m) that acts as a mapping relationship from m to o. W is built on the relationship: $\Delta m_t = 0.5(m_{t+1} - m_{t-1})$. More details on the definition of W can be found in [43].

The solution of maximizing the posterior probability $P(m|s)$ is equivalent to maximize $P(o|s)$ by considering the relationship between m_t and Δm_t. The solution is formulated as:

$$\begin{aligned}\frac{\partial log P(o|s)}{\partial m} &= \frac{\partial log P(Wm|s)}{\partial m} \\ &= \frac{\partial log \prod_{t=1}^{T} \mathcal{N}(o_t, \mu_t, \sigma_t^2)}{\partial m} \\ &= \frac{\partial \Sigma_{t=1}^{T} log \mathcal{N}(o_t, \mu_t, \sigma_t^2)}{\partial m}\end{aligned} \tag{23}$$

m can be solved by setting this equation to 0, which has been solved by Tokuda et al. [43].

We apply this algorithm to compute a clipped animation trajectory within a moving window with a fixed frame size. In our experiments, the window size is set to 5 frames and the moving step is 1 frame. After each computation for a window, the middle frame in the clipped animation is taken as the output motion position at the corresponding time step. It means that the output motion value at time t depends on the motion state at time t and those 4 motion states around time t.

Figure 6 shows the synthesis process of Step 2. As can be seen, the output animation trajectory is smooth thanks to 2 factors. The first one is the continuous mediate/hidden signals (the continuous-state sequence) inferred by Step 1. The second one is to take into account the role of motion velocity in motion trajectory, which is done by Step 2.

4.3 Shoulder Behavior Generator

As mentioned in Section 3, shoulder movement data is unavailable from our motion capture data. Hence, the statistical framework cannot be applied to synthesize shoulder animation. To overcome this lack of data we propose a rule-based method to produce shoulder animation from laughter audio signals. The mapping function from the audio feature, a_t, to shoulder movement value, m_t^{sld} , is defined as follows.

$$m_t^{sld} = \frac{1}{10} \Sigma_{t'=t}^{t-9} e^{(\frac{1}{2a_{t'}})} \tag{24}$$

As can be seen, m_t^{sld} is inferred from the audio features from $t-9$ to t. These were inferred from an empirical pre-study we conducted. The exponential

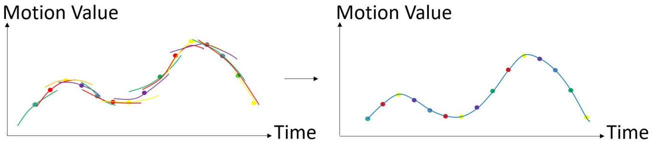


Figure 6: Trajectory synthesis in Step 2. The left figure shows a set of curves with different colors. Each curve is attached to a point of the same color. Each curve represents a clipped animation trajectory. It is synthesized from 5 neighboring motion states using Equation 23. Those points represent the calculated value for the middle time during 5 motion states. The right figure shows the curve following the points in the left figure. This curve is the output animation.

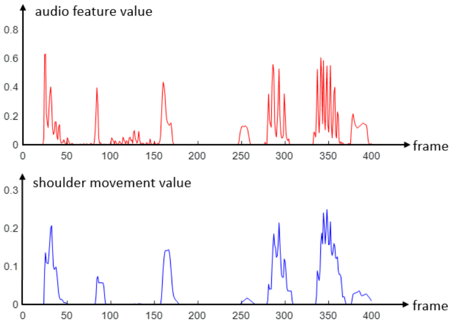


Figure 7: Example of shoulder animation according to Audio Feature. The upper curve is the input signal of audio feature; the bottom one is the output signal of shoulder Animation. The values in both curves are normalized between 0 and 1.

function is used to emphasize the audio feature with high value and to degrade the audio feature with low value. Figure 7 shows an example of the synthesized shoulder animations from the input audio features. As can be observed, the shoulder movement follows the audio feature, but no shoulder movement occurs when the audio signal has low values. When the audio feature trembles with high frequency, shoulder moves with high amplitude and embeds quick shaking. It is in line with Niewiadomski et al. [39] which reports that shoulder movements could be composed of two types of shaking movements with high and low frequencies.

5 Subjective Evaluation

To validate our laughter behavior controller, we investigate: (1) whether the continuous-state method outperforms the discrete-state method in head and torso animation synthesis; (2) whether the rule-based method for shoulder animations improves the perception of a laughing virtual character. These two investigations were conducted through a subjective evaluation.

The subjective evaluation was done through an online web application. Human participants were invited to watch clips of a laughing virtual character and then to evaluate the quality of its laughter behavior. The clips were obtained with our laughter behavior controller and with another state of the art model. We aim to compare our model with previous ones. So far, only [29] [38] focused on generating laughter torso and head animations. In [29], a rule-driven approach is used to synthesize torso animation. It is based on the hypothesis that torso movement always linearly follows the head movement, which is not fully verified in human data. In [38], a discrete-state approach is proposed to compute head and torso animation. It is based on an extension of standard HMM, called Coupled Transition Parameterized Loop HMM (CTPLHMM). To validate our approach, the discrete-state approach by [38] is taken as reference work.

Protocol. The participants were first invited to watch the clips displaying a laughing virtual character. Their task was to answer a few questions according to their perception of each animation of the laughing virtual character. Here are the elements of the protocol we follow for the perceptive evaluation study.

A. Participants: there were totally 61 participants, 34 males and 27 females, with age ranging from 18 to 63 years (M=31.23 years, SD=8.24 years)).

B. Stimuli: 9 episodes of human laughter audio were selected from the testing dataset. They include 3 short-duration samples with low intensity, 3 short-duration samples with high intensity and 3 long-duration samples. In the long duration samples, laughter audio is very complex; it is made of sequences of low intensity, high intensity, and even silence. The short-duration samples last between 4s and 6s; the long-duration samples last between 23s and 28s.

Each audio sample is used as the input to the laughter behavior controller. Then the output animations and the corresponding audio samples are used to drive the animation of the virtual character. Each animation of the laughing virtual character is stored in a video clip. Figure 8 shows three representative snapshots from the animation clips used in the subjec-

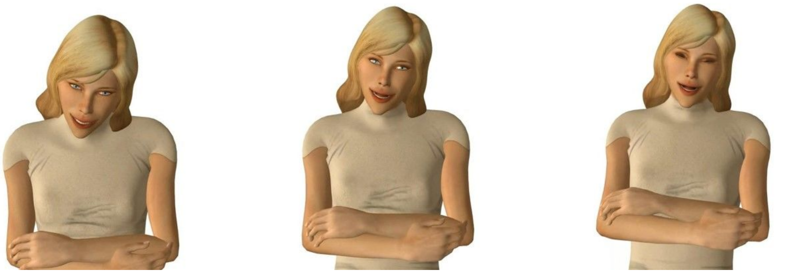

(a) Lean forward (b) Upright (c) Lean backward and tilt

Figure 8: Snapshots of synthesized laughter animation.

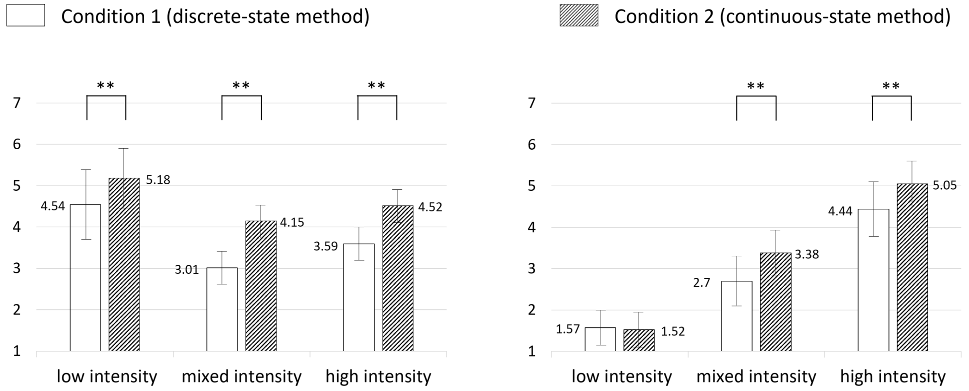
tive evaluation. Our work is dedicated to behaviors synthesis and not to the appearance of the virtual character. Therefore, the animations synthesized by the proposed and the reference methods are displayed by the same virtual agent (see Figure 8).

Three sets of animations of the virtual character were created for each selected audio sample. In all the animation clips, the virtual character's facial expressions are produced by our previous work [29]. The three sets of animations correspond to three conditions:

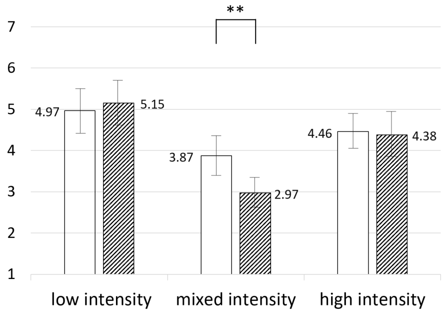
1. Condition 1: head and torso animations are generated by the reference method (the discrete-state method) [38];
2. Condition 2: head and torso animations are generated by our proposed method (the continuous-state method);
3. Condition 3: head, torso and shoulder animations are generated by our proposed method.

Therefore, there are a total of 27 animation clips (3 conditions × 9 laughter audio samples).

C. Design and Procedure: subjective evaluations were conducted online. First, each participant filled out a demographic questionnaire concerning their age, gender, education level, occupation and country in which participant spent the majority of his/her life. Then, the participant is invited to randomly watch 21 out of 27 stored animation clips. The 21 animation clips are comprised of all the 18 ones from Conditions 1 and 2 (2 conditions × 9 laughter audio samples) and 3 selected animation clips from Condition



(a) Degree of naturalness (b) Degree of amusement



(c) Degree of fakeness

Figure 9: Investigation on head and torso animations. The scores are the averaged values rated by participants in the subjective evaluation (standard deviation is shown in parenthesis). The statistically significant difference is estimated using independent samples t-test. ** marks p<.01.

3 containing a short-duration sample with low intensity, a short-duration sample with high intensity and a long-duration sample with mixed intensity. We chose to show to the participants only 3 animation clips in Condition 3 rather than all 9 ones to avoid demotivating participants over a too long experiment.

After watching each animation clip, the participant had to answer the following questions using a 7 point Likert scale:

1. How natural is the laughing behavior overall?
2. Is the virtual character freely expressing its amusement?
3. Is the virtual character faking laughter?

These questions are inspired by [49]. The participants were invited to quantify their perception of the laughter animation clips by answering the 3 questions above.

Hypothesis. The subjective evaluation aims to assess the following research hypotheses:

To investigate whether the continuous-state method outperforms the discrete-state method (the reference method) in head and torso animation synthesis. The following research hypothesis is assessed.

- **H1**: the virtual character with head and torso animations from the continuous-state method is perceived more natural (**H1-nat**), more amused (**H1-amu**) and displaying less fake laughter (**H1-fak**) than that from the discrete-state method [38].

To investigate whether the rule-based method for shoulder animations improves the perception of the laughing virtual character, the following research hypothesis is assessed.

- **H2**: the virtual character with the shoulder animations from our laughter behavior controller is perceived more natural (**H2-nat**), more amused (**H2-amu**), and displaying less fake laughter (**H2-fak**) than that with no shoulder animation.

6 Results

In this section, we report the results of the perceptual studies under the three conditions mentioned in Section 5.

Results on Head and Torso. To verify hypothesis H1, we study the differences in perception from the continuous-state method (described in this paper) and the discrete-state one [38]. The results from Conditions 1 and 2 are compared along with 3 factors: naturalness of animation, degree of amusement conveyed by the virtual character's behavior, and level of fake expression. The comparison is separately carried out for low intensity, mixed intensity, and high intensity. It involves only head and torso animations. Its results are evaluated by independent samples t-test. They are shown in Figure 9 and in Table 1. The results about naturalness, amusement and fakeness are reported as follows.

A. H1-nat. To verify hypothesis H1-nat, we compare the rated scores of answering the question of "How natural is the laughing behavior overall?" in the continuous-state method and the discrete-state method. The comparison results are shown in Figure 9a and Table 1.

Table 1: Comparison results between conditions 1 and 2 (discrete-state and continuous methods) from independent samples t-test on the perception of naturalness, amusement, and fakeness. ** marks p<.001.

	naturalness	amusement	fakeness
low intensity	t=-3.90 p<.001**	t=0.55 p<0.59	t=-1.31 p<0.19
mixed intensity	t=-8.79 p<.001**	t=-5.07 p<.001**	t=6.67 p<.001**
high intensity	t=-5.95 p<.001**	t=-3.75 p<.001**	t=0.60 p<0.55

As can be seen, significant differences are statistically observed in low, mixed and high intensities for the question about naturalness. The continuous-state method is quantified with higher values than the discrete-state method. It is also observed that the difference of mixed/high intensity between both methods is approximately 1 (1.14(=4.15-3.01) for mixed intensity and 0.93(=4.52-3.59) for high intensity) and the difference of low intensity is less than 1 (0.64=5.18-4.54). The differences of mixed intensity and high intensity are higher than that of low intensity. It suggests that the continuous-state method is capable of capturing more complex relationship than the discrete-state method and it is capable of rendering the captured relationship into the synthesized animations.

According to these results, Hypothesis H1-nat is verified in low intensity, mixed intensity, and high intensity.

B. H1-amu. To verify hypothesis H1-amu, we compare the rated scores of answering the question of ”Is the virtual character freely expressing its amusement?” in the continuous-state method and the discrete-state method. The comparison results are shown in Figure 9b and Table 1.

As can be seen, a significant difference is statistically observed in high/mixed intensity but no significant difference is found in low intensity. The continuous-state method is rated with the higher score than the discrete-state method. In both methods, it is observed that the highest values are observed in high intensity and that the values in mixed intensity are higher than those in low intensity.

According to these results, Hypothesis H1-amu is verified for laughter

Table 2: Pairwise comparisons using the Tukey HSD test on perception of shoulders. * marks p<.05 and ** marks p<.001.

	naturalness	amusement	fakeness
Conds 1 vs. 2	p<.001**	p=.003*	p=0.46
Conds 1 vs. 3	p<.001**	p<.001**	p<.001**
Conds 2 vs. 3	p<.001**	p<.001**	p<.001**

with high and mixed intensities but it is not verified for laughter with low intensity. It suggests that mixed-intensity or high-intensity animations from the continuous-state method are perceived as more amused than those from the discrete-state method.

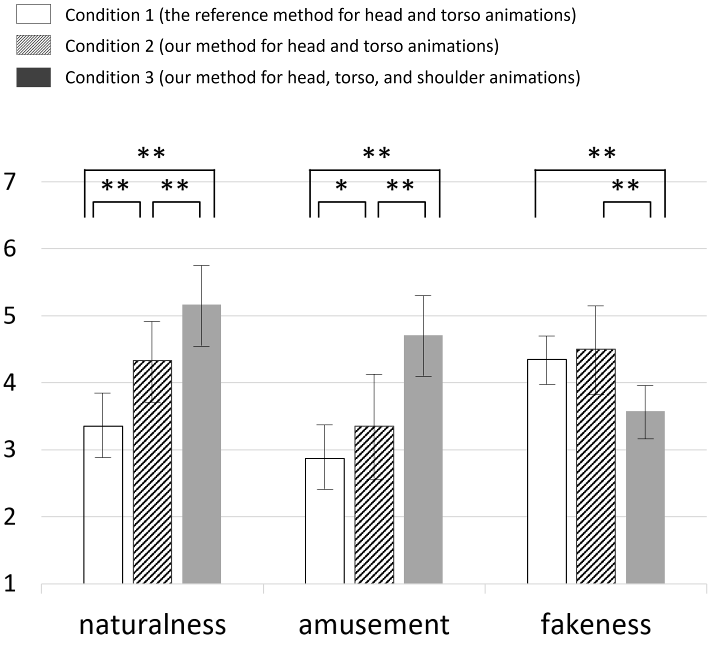
C. H1-fak. To verify hypothesis H1-fak, we compare the rated scores of answering the question ”Is the virtual character faking laughter?” in the continuous-state method and the discrete-state method. The comparison results are shown in Figure 9c and Table 1.

As can be seen, no significant difference between both conditions is statistically found for low and high intensities related to the degree of fakeness but a significant difference is observed for mixed intensity. The continuous-state method is rated with a lower score than the discrete-state method.

We can also observe that, for both methods, the rated values of low intensity is slightly higher than those of high intensity and that the rated values of low and high intensities are much higher than those of mixed intensity.

According to these results, Hypothesis H1-fak is verified only for laughter with mixed intensity and it is not verified for laughter with low intensity and high intensity. It suggests that the mixed-intensity laughter of the virtual character animated by the continuous-state method appears more authentic than the one computed by the discrete-state method.

Results on Shoulders. To verify hypothesis H2, we compare three conditions in terms of naturalness, amusement, and fakeness. In this comparison, laughter intensity is not a variable. To conduct the comparison among the three conditions, 3 animation clips, from each condition which uses the same 3 audio stimuli with mixed intensity, were selected and in-



(a) Degree of naturalness

Figure 10: Investigation on shoulder animation. The scores are the averaged values under Conditions 1, 2 and 3, which are rated by participants in terms of naturalness, amusement and fakeness (the standard deviation is shown in parenthesis). * and ** stand for the statistically significant difference between the discrete-state (reference) and continuous-state (proposed) methods. The statistically significant difference is estimated using independent samples t-test. * marks p<.05 and ** marks p<.001.

volved in this comparison. They differ only in the animation of the virtual character. The 61 participants evaluated all the clips.

Figure 9 shows the evaluated average scores. One-way ANOVA is applied to measure if there are significant differences between the three conditions in terms of naturalness, amusement, and fakeness, respectively. The test results show that there are statistically significant differences between the three conditions in terms of naturalness (F(2,546)=94.68, p<.001), amusement (F(2,546)=31.54, p<.001), and fakeness (F(2,546)=86.1, p<.001). To determine which specific conditions differ from each other, we carry out Tukey HSD (honest significant difference) post-hoc tests on all pairwise comparisons in terms of naturalness, amusement, and fakeness, respectively. The results of pairwise comparisons are shown in Figure 10 and Table 2.

Pairwise comparisons revealed that Conditions 1-2 and Condition 3 are statistically different along all three terms. The animations of Condition 3 are perceived as the most natural and amused, as well as the least fake (the most authentic).

According to the results above, Hypotheses H2-nat, H2-amu and H2-fak are verified in mixed intensity. It suggests that the produced shoulder animations make the laughing virtual character to appear more natural, amused and authentic.

7 Discussion

This section discusses the results of the subjective evaluation. First, it focuses on the results on head and torso animations in terms of naturalness, amusement, and fakeness; then it continues to discuss the results on shoulder animation.

A. Head and Torso: Perception of Naturalness.

The perceived level of naturalness in the animations of the virtual character appears much higher when animations are computed with the continuous-state method than with the discrete-state method. These results suggest that the continuous-state method is able to capture better the subtle dynamics of human laughter movements than the discrete-state method. The continuous-state method focuses not only on learning the correlation between laughter audio features and visual motions at the frame level but also on capturing dynamic movements between neighboring frames. On the other hand, the discrete-state method only focuses on extracting the coupling between the discrete intensities of laughter pseudo-phoneme and shaking motions types. In other words, the continuous-state method works at the level of frame, while the discrete-state method works at the level of laughter pseudo-phoneme. Working at the frame level ensures the continuous-state method to consider laughter intensity variations at a precise level and thus to render with more precision the laughter motion dynamics. The discrete-state method works at a coarser level, which results in degrading the animation quality.

The significant differences between both methods are much higher in the animations with high/mixed intensity than in the animations with low intensity. [49] suggests that low- intensity laughter audio is perceived as being natural when it is accompanied with low-intensity animations. The continuous-state and the discrete-state methods seem to both capture such a congruent coupling between laughter audio and animations with low intensity. This is not true for animations with high and mixed intensities where body motions vary a lot. In these cases, the animations computed with the continuous-state method appear more natural. They can render the dynamics of laughter motion with more accuracy increasing its quality.

B. Head and Torso: Perception of Amusement Level.

The results we obtained for stimuli with different intensities are in line with previous research. Indeed, Niewiadomski and colleagues [49] report that high-intensity laughter is considered to express amusement more freely than low-intensity laughter. It implies that low-intensity laughter may be perceived as corresponding to the low level of amusement, or even to no amusement. This could explain why in our perceptual study no significant difference between both methods was found for low-intensity animations. The authors also suggest that high-intensity laughter is often perceived as corresponding to the high level of amusement. This is also supported by our evaluation study which found that the animations with high and mixed intensities are perceived expressing amusement more freely than animations with low intensity.

When comparing the animations from both computational models, we find that mixed-intensity and high-intensity animations from the continuous-state method are capable of expressing amusement more freely than those from the discrete-state method. This observation suggests that the continuous-state method is more suitable to produce amusing laughter animations than the discrete-state method.

C. Head and Torso: Perception of Fake Expression.

The comparisons in term of fakeness show that no significant difference is observed for low and high intensities and that a small difference is observed for mixed intensity. This suggests that the continuous-state method has no obvious contribution to make the laughing virtual character appear more authentic than the discrete-state method. It could be explained since, when participants viewed a laughing character, they did not know why the character was laughing. No contextual information had been provided to participants. Not knowing why the agent laughed could enforce the impression of fake or exaggerated laughter. Moreover, participants heard identical laughter audio in both methods while observing different animations of the virtual character. The laughter audio could contribute more to the estimation of fake laughter than the visual animations.

From the results, the rated scores in low and high intensities are higher than the mean value (4.0); but the rated scores in mixed intensity is close to the mean value (3.5). These observations can be found for both methods. They are probably related to the length of laughter episodes. The laughter episodes in low and high intensities are relatively short about 4s-6s while the ones in mixed intensity last for 23s-28s. The laughter episodes with long duration provide more information to be perceived than the ones with short duration. With more information, the participants could perceive the laughter more reasonable and authentic (neither fake nor exaggerated).

It is also found that the continuous-state method leads to a lower score in term of fake laughter than the discrete-state method when more information is perceived in mixed intensity. Considering the results from head and torso animations in terms of naturalness, amusement, and fakeness, it suggests that more natural laughter animation is perceived more authentic and more freely expressing amusement.

D. Shoulders: Naturalness, Amusement and Fakeness.

The comparisons between the animations with and without shoulder animations show that significant differences are observed in terms of naturalness, amusement, and fakeness. It indicates that the shoulders play an important role in laughter perception. The results show that the animations with shoulder motions are perceived as more natural and more freely expressing amusement. Importantly, it is found that shoulder animations make the laughing virtual character less fake its laugh. These observations suggest that shoulder animation is effective. Shoulder animation is inferred from audio features at each frame. It not only reflects the dynamics of laughter audio but also avoids discontinuities.

According to this investigation on shoulder animation in terms of naturalness, amusement, and fakeness, we learn that laughter animation, when viewed as more natural, is perceived as expressing amusement more freely and appears as more authentic. This result is also found for laughter animations made of only head and torso motions.

8 Conclusion

In this paper, a laughter behavior controller is proposed to synthesize laughter animations of virtual characters upper body. The controller takes laughter audio signals as input. It performs at each time frame, based on the current audio input and the output inferred from the previous time frames.

To build the laughter behavior controller, a human laughter dataset was recorded. It contains laughter head and torso behaviors as well as laughter audio while shoulder movements are unavailable. The laughter behavior controller is comprised of two modules: one infers head and torso animations by a unified method, which is based on a statistical framework; the other one yields shoulder animations, which is based on a rule-based method.

We conducted a perceptive study to validate our laughter behavior controller. We compared animations obtained with our model, also referred to as a continuous-state method, with the state-of-the-art method taken as the reference method (the discrete-state method). The contribution of shoulder animations was measured by rating animations of the virtual character with and without shoulder animations. The animations were evaluated along with three factors: degree of naturalness, level of amusement and degree of fakeness. The evaluation results show that our method outperforms the reference method in terms of naturalness, amusement, and fakeness.

Considering the results on naturalness and amusement, it is observed that the laughter with the continuous-state method expresses amusement more freely than that with the discrete-state method; and it is also observed that our method is capable of producing animations where the character appears more natural and authentic than with the reference one. The module of generating shoulder animation contributes to the improvement of the quality of animation in the three factors.

To conclude, in this paper, we have presented a new statistical framework to learn the correlation of laughter behaviors and its audio. Our main contribution is to propose a continuous-state statistical framework which differs from the existing discrete-state statistical frameworks which are used to synthesize speech animations or laughter animations. The continuous-state framework is capable of capturing the laughter dynamics from human data, rendering the human motion dynamics into the generated animations, and directly generating smooth animation trajectories without any additional specific interpolation techniques.

One major limitation lies in that our laughter behavior controller is not compared with ground truth data. We could not perform such a comparison as our mocap dataset does not contain shoulder motion and it captures incomplete torso data. So the recorded mocap data is inadequate to display real human movements. In our pilot study, the animations produced by our method outperform the torso data collected by all three sensors placed along the torso, but it is inappropriate to report this observation as human data is not accurately captured (see Section 3).

In future work, motion tracking algorithms will be implemented to detect shoulder movements from the recorded laughter videos. This will allow us to develop a statistical framework to produce shoulder animations. In addition, as laughter can be related to multiple emotions (see Section Introduction), we are thinking about extending our approaches to yield laughter animations which convey various emotions. Since our laughter behavior controller works at the frame level, it could make real-time synthesis possible. It can be used to simulate a virtual character laughing while interacting with a human user.

References

[1] D. Morrison, R. Wang, and L. C. D. Silva, "Ensemble methods for spoken emotion recognition in call-centres," *Speech Communication*, vol. 49, no. 2, pp. 98 – 112, 2007.

[2] W. L. Chafe, *The importance of not being earnest: The feeling behind laughter and humor.* Amsterdam: John Benjamins Pub. Co, 2007.

[3] W. Ruch and P. Ekman, "The Expressive Pattern of Laughter," in *Emotion, qualia, and consciousness*, A. W. Kaszniak, Ed. World Scientific Publishers, Tokyo, 2001, pp. 426–443.

[4] R. Provine, "Laughter," *American Scientist*, vol. 84, no. 1, pp. 38–47, 1996.

[5] V. Adelsward, "Laughter and dialogue: The social significance of laughter in institutional discourse," *Nordic Journal of Linguistics*, vol. 102, no. 12, pp. 107–136, 1989.

[6] M. Owren and J. Bachorowski, "The evolution of emotional expression: a selfish-gene account of smiling and laughter in early hominids and humans," in *Emotion: Current Issues and Future Directions*, 2001, pp. 152–191.

[7] W. Ruch, G. Kohler, and C. Van Thriel, "Assessing the 'humorous temperament': Construction of the facet and standard trait forms of the state-trait-cheerfulness-inventory," *Humor: International Journal of Humor Research*, vol. 9, pp. 303–339, 1996.

[8] J. Foer, "Laughter: A scientific investigation," *The Yale Journal of Biology and Medicine*, vol. 74, no. 2, pp. 141–143, 2001.

[9] T. Huber and W. Ruch, "Laughter as a uniform category? A historic analysis of different types of laughter," in *In 10th Congress of the Swiss Society of Psychology.* University of Zurich, Switzerland, 2007.

[10] P. Ekman and W. Friesen, "Felt, false, miserable smiles," *Journal of Nonverbal Behavior*, vol. 6, no. 4, pp. 238–251, 1982.

[11] H. Griffin, M. Aung, B. Romera-Paredes, C. McLoughlin, G. McKeown, W. Curran, and N. Bianchi-Berthouze, "Laughter type recognition from whole body motion," in *Affective Computing and Intelligent Interaction*, 2013, pp. 349–355.

[12] H. J. Griffin, M. S. H. Aung, B. Romera-Paredes, C. McLoughlin, G. McKeown, W. Curran, and N. Bianchi-Berthouze, "Perception and automatic recognition of laughter from whole-body motion: Continuous and categorical perspectives," *IEEE Transactions on Affective Computing*, vol. 6, no. 2, pp. 165–178, 2015.

[13] R. Niewiadomski, M. Mancini, G. Varni, G. Volpe, and A. Camurri, "Automated laughter detection from full-body movements," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 1, pp. 113–123, 2016.

[14] M. Mancini, G. Varni, D. Glowinski, and G. Volpe, "Computing and evaluating the body laughter index," in *Proceedings of the Third International Conference on Human Behavior Understanding*, 2012, pp. 90–98.

[15] M. Mancini, L. Ach, E. Bantegnie, T. Baur, N. Berthouze, D. Datta, Y. Ding, S. Dupont, H. Griffin, F. Lingenfelser, R. Niewiadomski, C. Pelachaud, O. Pietquin, B. Piot, J. Urbain, G. Volpe, and J. Wagner, "Laugh when you're winning," in *Innovative and Creative Developments in Multimodal Interaction Systems.* Springer Berlin Heidelberg, 2014, vol. 425, pp. 50–79.

[16] M. Mancini, B. Biancardi, F. Pecune, G. Varni, Y. Ding, C. Pelachaud, G. Volpe, and A. Camurri, "Implementing and evaluating a laughing virtual character," *ACM Trans. Internet Technol.*, vol. 17, no. 1, pp. 3:1–3:22, Feb. 2017.

[17] G. McKeown, W. Curran, C. McLoughlin, H. J. Griffin, and N. Bianchi-Berthouze, "Laughter induction techniques suitable for generating motion capture data of laughter associated body movements," *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pp. 1–5, 2013.

[18] M. Chollet, M. Ochs, and C. Pelachaud, "From non-verbal signals sequence mining to Bayesian networks for interpersonal attitudes expression," in *Intelligent Virtual Agents*, 2014, pp. 120–133.

[19] T. Bickmore, L. Bukhari, L. P. Vardoulakis, M. Paasche-Orlow, and C. Shanahan, "Hospital buddy: A persistent emotional support companion agent for hospital patients," in *Intelligent Virtual Agents*, 2012, pp. 492–495.

[20] L. P. Vardoulakis, L. Ring, B. Barry, C. L. Sidner, and T. Bickmore, "Designing relational agents as long term social companions for older adults," in *Intelligent virtual agents*, 2012, pp. 289–302.

[21] S. Kopp, L. Gesellensetter, N. C. Krämer, and I. Wachsmuth, "A conversational agent as museum guide – design and evaluation of a real-world application," in *Intelligent Virtual Agents*, 2005, pp. 329–343.

[22] Y. Ding, M. Radenen, T. Artières, and C. Pelachaud, "Speech-driven eyebrow motion synthesis with contextual Markovian models." in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 3756–3760.

[23] Y. Ding, C. Pelachaud, and T. Artières, "Modeling multimodal behaviors from speech prosody," in *Intelligent Virtual Agents*, 2013, pp. 217–228.

[24] J. Cassell, H. Vilhjálmsson, and T. Bickmore, "BEAT : the Behavior Expression Animation Toolkit," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, pp. 477–486.

[25] E. Bevacqua, K. Prepin, R. Niewiadomski, E. de Sevin, and C. Pelachaud, "Greta: Towards an interactive conversational virtual companion," *Artificial Companions in Society: perspectives on the Present and Future*, pp. 1–17, 2010.

[26] E. Bozkurt, E. Erzin, and Y. Yemez, "Affect-expressive hand gestures synthesis and animation," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6.

[27] 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*, 2015, pp. 139–151.

[28] Y. Ding, L. Shi, and Z. Deng, "Perceptual enhancement of emotional mocap head motion: An experimental study," in *International Conference on Affective Computing and Intelligent Interaction*, 2017. To appear.

[29] Y. Ding, K. Prepin, J. Huang, C. Pelachaud, and T. Artières, "Laughter animation synthesis," in *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, 2014, pp. 773–780.

[30] F. Pecune, M. Mancini, B. Biancardi, G. Varni, Y. Ding, C. Pelachaud, G. Volpe, and A. Camurri, "Laughing with a virtual agent," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems.* International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 1817–1818.

[31] F. Pecune, B. Biancardi, Y. Ding, C. Pelachaud, M. Mancini, G. Varni, A. Camurri, and G. Volpe, "Lol-laugh out loud." in *AAAI*, 2015, pp. 4309–4310.

[32] H. Çakmak, J. Urbain, J. Tilmanne, and T. Dutoit, "Evaluation of HMM-based visual laughter synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014, pp. 4578–4582.

[33] Y. Ding and C. Pelachaud, "Lip animation synthesis: a unified framework for speaking and laughing virtual agent," in *Auditory-Visual Speech Processing*, 2015, pp. 78–83.

[34] H. Cakmak, K. E. Haddad, and T. Dutoit, "GMM-based synchronization rules for HMM-based audio-visual laughter synthesis," in *International Conference on Affective Computing and Intelligent Interaction*, 2015, pp. 428–434.

[35] D. Cosker and J. Edge, "Laughing, crying, sneezing and yawning: Automatic voice driven animation of non-speech articulations," *Proceedings of Computer Animation and Social Agents*, pp. 21–24, 2009.

[36] R. Niewiadomski and C. Pelachaud, "Towards multimodal expression of laughter," in *International Conference on Intelligent Virtual Agents*, 2012, pp. 231–244.

[37] P. C. DiLorenzo, V. B. Zordan, and B. L. Sanders, "Laughing out loud: Control for modeling anatomically inspired laughter using audio," in *ACM SIGGRAPH Asia*, 2008, pp. 125:1–125:8.

[38] Y. Ding, J. Huang, N. Fourati, T. Artières, and C. Pelachaud, "Upper body animation synthesis for a laughing character," in *Intelligent Virtual Agents*, 2014, vol. 8637, pp. 164–173.

[39] R. Niewiadomski, M. Mancini, Y. Ding, C. Pelachaud, and G. Volpe, "Rhythmic body movements of laughter," in *Proceedings of the 16th International Conference on Multimodal Interaction*, 2014, pp. 299–306.

[40] R. Niewiadomski, J. Hofmann, J. Urbain, T. Platt, J. Wagner, B. Piot, H. Cakmak, S. Pammi, T. Baur, S. Dupont, M. Geist, F. Lingenfelser, G. McKeown, O. Pietquin, and W. Ruch, "Laugh-aware virtual agent and its impact on user amusement ," in *International Conference on Autonomous Agents and Multiagent Systems*, 2013, pp. 619–626.

[41] J. Urbain, E. Bevacqua, T. Dutoit, A. Moinet, R. Niewiadomski, C. Pelachaud, B. Picart, J. Tilmanne, and J. Wagner, "The AVLaughterCycle database," in *Language Resources and Evaluation Conference*, 2010, pp. 2996–3001.

[42] H. Çakmak, J. Urbain, and T. Dutoit, "Synchronization rules for HMM-based audio-visual laughter synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 2304–2308.

[43] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2000, pp. 1315–1318.

[44] M. Brand, "Voice puppetry," in *Proceedings of conference on Computer graphics and interactive techniques*, 1999, pp. 21–28.

[45] F. Pecune, A. Cafaro, M. Chollet, P. Philippe, and C. Pelachaud, "Suggestions for extending saiba with the vib platform," in *Proceedings of the Workshop on Architectures and Standards for Intelligent Virtual Agents, Intelligent Virtual Agents*, 2014.

[46] I. S. Pandzic and R. Forchheimer, Eds., *MPEG-4 Facial Animation: The Standard, Implementation and Applications.* New York, NY, USA: John Wiley & Sons, Inc., 2003.

[47] P. Boersma and D. Weeninck, "PRAAT, a system for doing phonetics by computer." *Glot International*, vol. 5, no. 9/10, pp. 341–345, 2001.

[48] D. H. Eberly, *3D Game Engine Design, Second Edition: A Practical Approach to Real-Time Computer Graphics (The Morgan Kaufmann Series in Interactive 3D Technology).* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2006.

[49] R. Niewiadomski, Y. Ding, M. Mancini, C. Pelachaud, G. Volpe, and A. Camurri, “Perception of intensity incongruence in synthesized multimodal expressions of laughter,” in *Affective Computing and Intelligent Interaction*, 2015, pp. 684–690.

[ ]Yu Ding is a postdoctoral researcher at the University of Houston, Texas, USA. He completed Computer Science Ph.D. (2014) at Telecom Paristech in Paris, France. He received his M.S. degree in Computer Science from Pierre et Marie Curie university in Paris, France and B.S. degree in automation from Xiamen University in Xiamen, China. His research interests include nonverbal communication (face, gaze, and gesture), expressive behaviors and human interaction.

[ ]Jing Huang is an associate professor at Zhejiang Gongshang Univeristy, School of Information and Electronic Engineering, and as co-founder of matrixvis co.ltd in Hangzhou, China. He was previously a contracted researcher at CNRS in the LTCI laboratory and Telecom ParisTech. His research interests are mainly interactive rendering and animation generation with parallel high-performance computing. He is also working on learning model in interactive communicative system. He obtained his PhD degree (2013) in computer graphics at TELECOM ParisTech, and received his master degree (2009) at University of Paris Descartes.

[ ]Catherine Pelachaud is a director of research at ISIR-CNRS, Université Pierre et Marie Curie, Paris, France. Her research interests include embodied conversational agent, nonverbal communication (face, gaze, and gesture), expressive behaviors, and socio-emotional agents.