Laughter Animation Synthesis

Yu Ding
Institut Mines-Télécom
Télécom Paristech
CNRS LTCI
Paris, France

Ken Prepin
Institut Mines-Télécom
Télécom Paristech
CNRS LTCI
Paris, France

Jing Huang
Institut Mines-Télécom
Télécom Paristech
CNRS LTCI
Paris, France

Catherine Pelachaud
Institut Mines-Télécom
Télécom Paristech
CNRS LTCI
Paris, France

Thierry Artières
Université
Pierre et Marie Curie
LIP6
Paris, France

ABSTRACT

Laughter is an important communicative signal in human-human communication. However, very few attempts have been made to model laughter animation synthesis for virtual characters. This paper reports our work to model hilarious laughter. We have developed a generator for face and body motions that takes as input the sequence of pseudo-phonemes of laughter and each pseudo-phoneme's duration time. Lip and jaw movements are further driven by laughter prosodic features. The proposed generator first learns the relationship between input signals (pseudo-phoneme and acoustic features) and human motions; then the learnt generator can be used to produce automatically laughter animation in real time. Lip and jaw motion synthesis is based on an extension of Gaussian Models, the contextual Gaussian Model. Head and eyebrow motion synthesis is based on selecting and concatenating motion segments from motion capture data of human laughter while torso and shoulder movements are driven from head motion by a PD controller. Our multimodal behaviors generator of laughter has been evaluated through perceptive study involving the interaction of a human and an agent telling jokes to each other.

Categories and Subject Descriptors

H.5.1 [**Multimedia Information Systems**]: Animations and Artificial, augmented, and virtual realities

General Terms

Algorithms, Human Factors, Experimentation

Keywords

multimodal animation, expression synthesis, laughter, virtual agent

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

1. INTRODUCTION

Laughter is an essential communicative signal in human-human communication: it is frequently used to convey positive information about human affects; it can be used as feedbacks to humorous stimuli or praised statements; it can be used to mask embarrassment; it can act as social indicator of in-group belonging [1]; it can play the role of speech regulator during conversation [17]. Laughter may also have positive effects on health [9]. Laughter is extremely contagious [17] and can be used to elicit interlocutor's laughter.

Our aim is to develop an embodied conversational agent able to laugh. Laughter is a multimodal process involving speech information, facial expression and body gesture (e.g. shoulders and torso movements), which often occurred with observable rhythmicity [18]. Niewiadomski and Pelachaud [12] indicated that the synchronization among all the modalities is crucial for laughter animation synthesis. Humans are very skilled in reading nonverbal behaviors and in detecting even small incongruences in synthesized multimodal animations. Embodied conversational agents ECAs are autonomous virtual agents able to converse with human interactants. As such their communicative behaviors are generated in real-time and cannot be pre-stored. To achieve our aim to simulate laughing agent, we ought to reproduce the multimodal signals of laughter and their rhythmicity. We have developed a multimodal behaviors synthesis for laughter based on motion capture data and on a statistical model.

At a first stage, we focus on hilarious laughter that is laughter triggered from amusing and positive stimuli (e.g., a joke). We use the AVLaughterCycle database [21] which contains motion capture data of the head movements and facial expressions of humans watching funny movies.

Our model takes as input the laughter segmentation in small sound units, called pseudo-phonemes [22] in reference to phonemes in speech, and their duration. Using audio-visual data of laughter, the model learns the correlation between lip data and these pseudo-phonemes. Due to the strong correlation between acoustic features (such as energy and pitch) and lip shape, our model considers also these features in computing lip shapes and jaw movement.

On the other hand, we do not consider speech features when computing head movements and facial expressions;

we keep only the pseudo-phonemes data. Indeed, many of the pseudo-phonemes in a laughter correspond to unvoiced speech, also called silent laughter [22]. Laughter intensity may be very strong even during these unvoiced segments [13]. Niewiadomski and Pelachaud [12] reported that there is a strong relationship between laughter behaviors and laughter intensity. Laughter with high intensity involves not only movements with larger amplitude but also different types of movement. For example, frown arises very often when the laugh is very strong but not when it is of low intensity. So instead of using speech features that can not capture these features (linked to silent laughter and laughter intensity), a cost function has been defined to select and concatenate head and eyebrows segments motion stored in motion capture database. Thus, our model accounts only on pseudo-phonemes for head movements and facial expressions.

The AVLaughterCycle database contains only data on head movements and facial expressions. Torso and shoulders movement has not been recorded using motion capture data. To overcome such missing data, we have built a controller linking torso movement and head one. We rely on observational study of the videos of the AVLaughterCycle database.

In the remaining of this paper we first describe related works in section 2. Then we describe the dataset used in our experiments in section 3 and we detail our multimodal motion synthesis in section 4. Finally we describe in details our experiments and we comment the results in section 5.

2. RELATED WORKS

In this section, we present related works on laughter motion synthesis.

DiLorenzo et al. [5] proposed a physics-based model of human chest deformation during laughter. This model is anatomically inspired and synthesizes torso muscle movements activated by the air flow within the body. Yet, the animation cannot be synthesized in real-time and the model can not be easily extended to facial motion (e.g. eyebrow) synthesis.

Cosker and Edge [4] used HMM to synthesize facial motion from audio features (MFCC). The authors built several HMMs to model laughter motion, one HMM per subject. To compute the laughter animation of new subject, the first step is to classify the laughter audio into one HMM by comparing the mostly likelihood. Then the selected HMM is used to produce the laughter animation. The authors do not precise how many HMMs should be built to cover various audio patterns from different subjects. The use of the classification operation as well as of the Viterbi algorithm makes impossible to obtain animation synthesis in real time. In the states sequence computed by the Viterbi algorithm, one single state may last very long. It leads to unchanged motion position during such a state, which produces unnatural animations.

Niewiadomski and Pelachaud [12] consider how laughter intensity modulates facial motion. A specific threshold is defined for each key point. Each key point moves linearly according to the intensity if it is higher than the corresponding threshold. So, if the intensity is high, the facial key points concerning laughter move more. In this model, facial motion position depends only on laughter intensity. It lacks of variability. Moreover, all facial key points move always synchronously, while human laughter expressions do not. For example, for the same intensity, one human subject can move both eyebrows, another one only one eyebrow. In their perceptive study, each laughter episode is specified with a single value of intensity. It leads to only one invariable facial expression during this laughter episode.

Later on, Niewiadomski et al. [11] propose an extension of their previous model. Recorded facial motion sequence is selected by taking into account two factors: laughter intensity and laughter duration. In this model, coarticulation of lip shapes is not considered which may lead to non-synchronisation between lip shape and audio information (e.g. closed lip and strong intensity audible laughter information). Moreover, the roles of intensity and duration are not attentively distinguished when selecting recorded motion sequence. As a side effect, the selected motion may last differently (e.g. too short) than the desired duration.

Urbain et al. [21] proposed to compare the similarity of new and recorded laughter audio information and then to select the corresponding facial expressions sequence. The computation of the similarity is based on the mean and standard deviation of each audio feature during the laughter audio sequence. It means that the audio sequence is specified by only two variables: mean and standard deviation. This is not enough to characterize long audio sequence.

3. DATABASE

Our work is based on the AVLaughterCycle database [21]. This database contains more than 1000 audiovisual spontaneous laughter episodes produced by 24 subjects. 66 facial landmarks coordinates were detected by an open-source face tracking tool - FaceTracker [19]. Among these 66 landmarks, 22 landmarks correspond to the Facial Animation Parameters FAPs of MPEG-4 [15] for the lips and 8 landmarks for the FAPs for the eyebrows.

In this database, subjects are seated in front of a PC and a set of 6 cameras. They watch funny movies for about 15mn. Their facial expressions, head movements and laughter are then analyzed using FaceTracker. However body behaviors (e.g. torso and shoulders behaviors) are not recorded in this database. 24 subjects were recorded but only 4 subjects had their head motion tracked. Therefore, a sub dataset of 4 subjects with head motion data is used in our work.

This database includes acoustic data of laughter. In particular it contains the segmentation of laughter into small sound units. [22] has categorized audible information from laughter into 14 pseudo-phonemes according to human hearing perception. These 14 pseudo-phonemes correspond to (number of occurrences of these pseudo-phonemes are specified in parentheses): silence(729), ne(105), click(27), nasal(126), plosive(45), fricative(514), ic(162), e(87), o(15), grunt(24), cackle(10), a(144), glotstop(9) and vowel(0). So laughter is segmented into sequences of pseudo-phonemes and their durations. Laughter prosodic features (such as energy and pitch) have also been extracted using PRAAT [2] and are provided with the database.

In our model we focus on face and head motion synthesis from laugher pseudo-phonemes sequence (e.g. [a, silence, nasal]) and their duration (e.g. [0.2s, 0.5s, 0.32s]). We take prosodic features as additional inputs for lip and jaw motion synthesis. Section 4 provides further details on our model.

Since the AVLaughterCycle database does not contain any annotation about torso movement, neither from sensors nor from analysis, we base our torso animation model on the

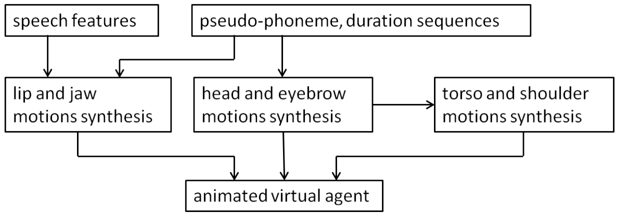

speech features
pseudo-phoneme, duration sequences
lip and jaw motions synthesis
head and eyebrow motions synthesis
torso and shoulder motions synthesis
animated virtual agent

Figure 1: Overall architecture of multimodal behaviors synthesis

observations that head and torso movements are correlated. As explained in section 4.3, we build a PD controller that extrapolates torso movement from head one.

4. MOTION SYNTHESIS

Figure 1 illustrates the overall architecture of our multimodal behavior synthesis. Our aim is to build a generator of multiple outputs (lip, jaw, head, eyebrow, torso and shoulder motions) from an input sequence of pseudo-phonemes together with their duration and from speech prosodic features (i.e. pitch and energy). Although one could consider designing a model that jointly synthesizes all the outputs from the inputs we use three different systems to synthesize three kinds of outputs. We briefly motivate our choices then we present in details the three modules.

First to accurately synthesize lip and jaw motions, which play an important role in articulation, we exploit all our inputs, namely the speech features and the pseudo-phoneme sequence, in a new statistical model that we describe in section 4.1. Using speech features as input yields an accurate synthesized motion that is well synchronized with speech, which is required for high quality synthesis.

Second, although it has been demonstrated in the past that speech features allow accurate prediction of head and eyebrow motion for normal speech [3, 8, 7, 6], the relationship between speech features and a laughter's head and eyebrow motion is unknown. Moreover exploring our laughter dataset we found that some segments have significant head and eyebrow motion while they are labeled as unvoiced segments. We then turned to exploit a more standard *synthesis by concatenation* method that we simplify to allow real time animation. Our method is described in section 4.2.

At last, body (torso and shoulder) motion, which are important components for laughter realism [18], are determined in a rather simple way from the synthesized head motion output by the algorithm in section 4.2. The main reason for doing so is that there is no torso and shoulder motion information gathered in our dataset so that none of the two synthesis methods above may be used here. Moreover we noticed in our dataset a strong correlation between head move on the one hand and torso and shoulders moves on the other hand. We then decided to hypothesize a simple relationship between the two motions that we modeled with a proportional-derivative (PD) controller. We present such a model in section 4.3.

4.1 Lip and jaw synthesis module

To design the lip and jaw motion synthesis system, we used what we call a contextual Gaussian model standard (CGM). A CGM is a Gaussian distribution whose parameters (we considered the mean vector but one could consider the covariance matrix as well) depend on a set of contextual variable(s) grouped in a vector θ (it is a vector of dimension c). Basically the underlying idea of a CGM is to estimate the distribution of a desired quantity x (the lip and jaw motion) as a function of an observed quantity θ (the speech features). In a CGM with a parameterized mean vector, the mean of the CGM obeys:

$$\hat{\mu}(\theta) = W^{\mu}\theta + \bar{\mu}_j \tag{1}$$

$$p(x|\theta) = N(x; \mu(\theta), \Sigma) \tag{2}$$

where W^{μ} is a $d \times c$ matrix, and $\bar{\mu}$ is an offset vector. θ stands for the value of contextual variable. This modeling is inspired from ideas in [7] where it has been shown to be accurate to predict motion from speech in normal speech situation.

We use one such CGM for each of the 14 pseudo-phonemes so that we get a set of 14 CGMs. Somehow, it is a conditional mixture of Gaussian distribution. Each model CGM is learned to model the dependencies between the lip/jaw motion and the speech features from a collection of training pairs of speech features and of lip and jaw motion.

The CGM model of a pseudo-phoneme is learned through Maximum Likelihood Estimation (MLE). For compact notation, we first define the matrix $Z^{\mu} = [W^{\mu} \;\; \bar{\mu}]$ and the column vector $\Omega_t = [\theta_t \;\; 1]^T$. Equation 1 can then be rewritten as $\hat{\mu}(\theta_t) = Z^{\mu} \times \Omega_t$. The solution of the MLE estimation may be easily found to be:

$$Z^{\mu} = [\sum_t x_t \Omega_t][\sum_t \Omega_t \Omega_t]^{-1} \tag{3}$$

where we consider a single training sequence case and the sum ranges over all indices in the sequence.

At synthesis time one has as inputs a series of speech features and a sequence of pseudo-phonemes together with their duration. The synthesis of the lip and jaw motion is performed independently for every segment corresponding to a pseudo-phoneme of the sequence then the obtained signal is smoothed at articulation between successive pseudo-phonemes. One can adopt few techniques to synthesize the lip and jaw motion segment given a pseudo-phoneme (with a known duration) and speech features.

A first technique consists in relying on a synthesis method that has been proposed for Hidden Markov Models by [20] which yields smooth trajectories. Alternatively, a simpler approach consists in using the speech features θ_t at time t to compute the most likely lip and jaw motion, i.e. $\mu(\theta_t)$. This is the approach we used in our implementation to ensure real time synthesis. Note that the obtained motion sequence $(\mu(\theta_t))_t$ is reasonably realist since speech features most often evolve smoothly.

4.2 Head and eyebrow synthesis module

Our approach to head and eyebrow synthesis system is based on selecting and concatenating motions from original data corresponding to the input pseudo-phonemes sequence. This may be done provided one has a large enough collection of real motion segments corresponding to every pseudo-phoneme. Such data are available from from the AVLaughterCycle database [21] which includes head and eyebrow motion data and which has been manually labeled

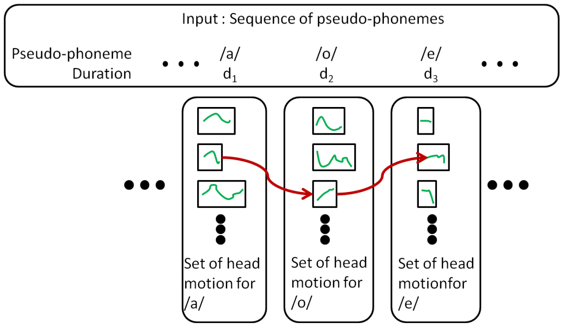

Input : Sequence of pseudo-phonemes
Pseudo-phoneme
Duration
/a/
d1
/o/
d2
/e/
d3
Set of head motion for /a/
Set of head motion for /o/
Set of head motionfor /e/

Figure 2: Head and eyebrow synthesis framework is performed by the concatenation of motion segments, gathered from real data, corresponding to a given pseudo-phoneme sequence and their duration. Green curve are samples of motion segments while the red arrow indicates the sequence of selected motion segments. The chosen motion segment is the one that minimizes a cost function of fit with the sequence of pseudo-phonemes.

into pseudo-phoneme segments. Actually for each of the 14 pseudo-phoneme labels, pp_i, we have a number N_i of head and eyebrow real moves that we note $S_i = \left\{m^i_j, j = 1..N_i\right\}$.

For a given pseudo-phoneme sequence of length K, $(p_1, ...p_K)$ (with $\forall k \in 1..K, p_k \in \{pp_1, ..., pp_{14}\}$), noting $d(p_k)$ the duration of the k^{th} pseudo-phoneme in the sequence, the *synthesis by concatenation* method aims at finding the best sequence of segments $(s_1, s_2, ..., s_K)$ belonging to $S_{p_1} \times S_{p_2} \times ... \times S_{p_K}$ (with $d(s_k)$ the duration of the segment) such that a cost function (that represents the quality of fit between the segment sequence and the pseudo-phonemes sequence) is minimized. Figure 2 illustrates our head and eyebrow synthesis framework. In our case the cost function is defined as:

$$C\left[(s_1, s_2, ..., s_K), (p_1, p_2, ..., p_K)\right] \tag{4}$$

$$= \gamma \sum_{u=1..K} C_{Dur}(d(s_u), d(p_u)) \tag{5}$$

$$+ (1-\gamma) \sum_{u=2..K} C_{Cont}(s_{u-1}, s_u) \tag{6}$$

where C_{Dur} is a *duration* cost function that increases with the difference between the length of a segment and the length of the corresponding pseudo-phoneme, and where C_{Cont} is a *continuity* cost function that increases with the distance between the last position of a segment and the first position of the following segment, and where γ is a manually tuned parameter (between 0 and 1) that allows weighting the importance of continuity and duration costs.

The two elementary cost functions are defined as follows, there are illustrated in Figure 3:

$$C_{Dur}(d, d') = e^{|d-d'|} - 1 \tag{7}$$

and:

$$C_{Cont}(s, s') = \left\|last(s) - first(s')\right\|^2 \tag{8}$$

where $first(s)$ and $last(s)$ stand for the first and the last positions in segment s.

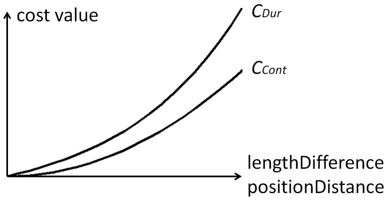


Figure 3: Shape of the duration cost function $C_{Dur} = f(v) = e^v - 1$ and of the continuity cost function $C_{Cont} = g(v) = v^2$ as a function of their argument v.

Once a sequence of segments $(s_1, s_2, ..., s_K)$ has been determined the synthesis of head and eyebrow motion corresponding to the pseudo-phonemes sequence requires some processing. Indeed the selected segments' duration may not be exactly the same as the pseudo-phonemes' duration. Selected segments are then linearly stretched or shrank to obtain the required duration. Note that it is assumed that stretching and shrinking of segment motion have no effect on human perception as long as segment duration has minimal variation. Also it may happen that there is a significant distance between the last frame of a segment and the first frame of the next segment which would yield discontinuous moves. To avoid this we perform a local smoothing by linear interpolation at the articulation between two successive segments.

Note that to allow real-time animation, we use a simplified version of the *synthesis by concatenation* method by selecting iteratively the first segment, then the second, then the third according to a *local* cost function focused on the current segment s, $\gamma C_{Dur}(d(s), d(p)) + (1-\gamma)C_{Cont}(s', s)$ where p stands for the current pseudo-phoneme, whose duration is $d(p)$, and s' stands for the previous segment. The obtained sequence of segments may then not be the one that minimizes the cost in Eq. (4), it is an approximation of it.

Note finally that the duration cost increases much quicker than the continuity cost (see Figure 3), which is wanted since as we said previously stretching and shrinking are tolerable only for small factors, while smoothing the end of a segment and the beginning of the following segment is always possible to avoid discontinuous animation. Defining the cost functions as in equations (7) and (8) strongly discourages high stretching and shrinking factors.

4.3 Torso and shoulder synthesis module

As we explained before torso and shoulder motion is synthesized from the synthesized head motion which is output by the algorithm described in the previous section. Although [18] reported torso and shoulders motions are important components of laughter, there is no such motion data in the AVlaughtercycle corpus. Thus the synthesis methods used for lip and jaw or for head and facial expressions cannot be used. Through careful observation of the AVlaughtercycle dataset we notice a strong correlation between torso and head movements. For instance we did not find any case where torso and head are going in opposite direction. Thus we hypothesize that torso and shoulder motion follows head motion and that a simple prediction module may already perform well for natural-looking animation.

Based on these observations, torso and shoulder movements of the virtual agent are synthesized from head movements. In more details, we define a desired intensity (or amplitude) of each torso and shoulder movement which is decided by the head movement. This desired intensity is the desired value in a PD (proportional derivate) controller. We choose to use a PD controller (illustrated in Fig 4) since it is widely used in graphics simulation domain [10], which is a simple version of proportional-integral-derivative controller (PID) in classical mechanics. The PD controller ensures smooth transitions between different motion sequences and removes discontinuity artifacts.

The PD controller is defined as:

$$\tau = k_p(\alpha_{current} - \alpha) - k_d\overline{\alpha}$$

where τ is the torque value, k_p is the proportional parameter, $\alpha_{current}$ is the current value of the head pitch rotation (ie vertical rotation as in head nod), α is the previous head pitch rotation, k_d is the derivative parameter, $\overline{\alpha}$ is the joint angle velocity. At the moment, we defined manually, by trial and error, the parameters of the PD controller.

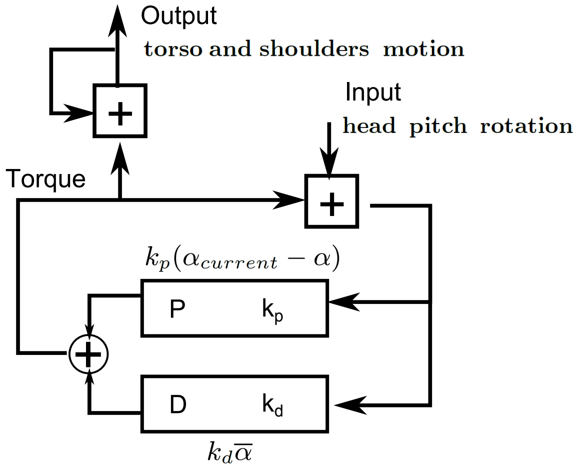


Figure 4: PD controller is used to compute torso and shoulders motion for each frame. Input: current head pitch rotation; Output: torso and shoulders joints

We define two controllers, one for torso joints (vt3, vt6, vt10, vl2) and one for shoulders joints (acromioclavicular, sternoclavicular) which are defined in MPEG4 H-ANIM skeleton [15]. The other torso joints are extrapolated from these 4 torso joints. To avoid any "freezing" effect we add a Perlin noise [16] on the 3 dimensions of the predicted torso joints.

Our PD controllers communicate with our laughter realizer module to generate laughter upper body motions. The laughter realizer module is used to synchronize all the laughter motions.

5. EXPERIMENTS

In this section we describe examples of laughter animations. We also present an evaluation study where the agent and human participants exchange riddles. The input to our motion synthesis model includes laughter pseudo-phonemes sequence, each phoneme duration and audio features (pitch and energy) sequence. Our motion synthesis model generates multimodal motions synchronized with laughter audio in real time. Figure 5, Figure 6 and Figure 7 present several frames of the animation synthesized by our approach.

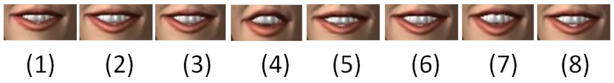


Figure 5: Synthesized lip, front view

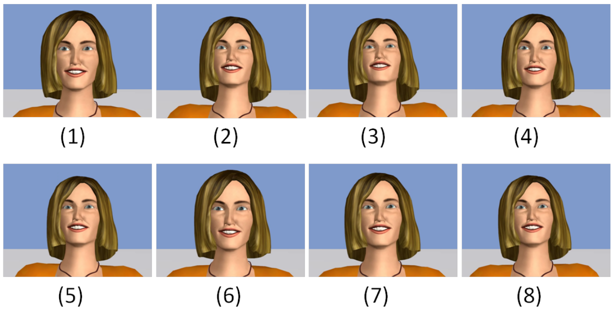


Figure 6: Synthesized data, front view

Our next step is to measure the effect of these laughs on partners of an interaction with a laughing agent. For this purpose, we have conducted a study to test how users perceive laughing virtual characters when the virtual character laughs during its speaking turn and when it listens. This study has been thought as a step further of Ochs and Pelachaud's study on smiling behaviour [14] (see below for a short description): the smiling behaviours used in [14] are used as the control condition; that is the virtual character smiles instead of laughing.

Considering the type of behaviour that we want to test, i.e. laugh, the experimental design of [14] is particularly appropriate. Indeed, in order to explore the effect of amusement smiling behaviours on users' perception of virtual agents, the authors chose positive situations to match the types of smile: in their experiment, the agent asks a riddle to the users, make a pause and give the answer. We use the four jokes and the description of polite and amused smiles of [14]'s evaluation study.

We have conducted a perceptive study to evaluate how users perceive how a virtual character laughs or smiles when, either telling a riddle, or listening to a riddle. We consider the following conditions: when the virtual character tells the joke and laughs or smiles, and when the human user tells the joke and the virtual character laughs or smiles.

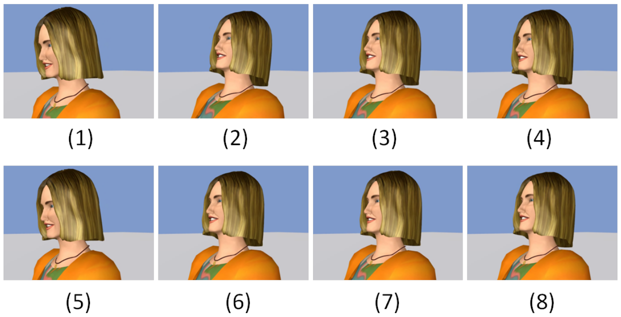


Figure 7: Synthesized data, side view

Thus, we have two ”test conditions” which are the laughing conditions, when speaking or listening, and two ”control conditions” which are the smiling conditions, when speaking or listening.

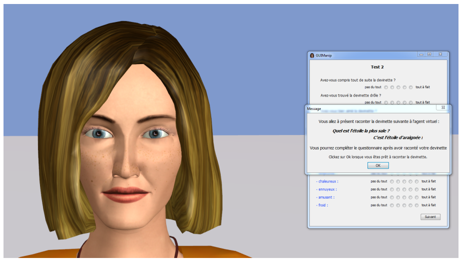
Hypotheses.

Our hypotheses are: (1) the evaluation of the agent’s attitude: we expect that the agent which laughs when the human user tells a joke will be perceived as warmer, more amused, more positive than the agent which only smiles; (2) the evaluation of the joke: we expect that when the agent laughs to the user’s joke, the user will evaluate “his” joke as funnier.

5.1 Setup

The main constraint for our evaluation is to have real time reaction of the agent to the human user’s behaviour. This constraint is induced by the listening agent condition in which the user tells the joke and the agent has to react at appropriate time, i.e. at the end of the joke. As a consequence for the design of our study, we cannot use pre-recorded videos of the agent’s behaviour and thus, we cannot perform the evaluation on the web as in [14]. We performed the evaluation in our lab.

Participants sit on a chair in front of computer screen. They wear headphones and microphone and have to use the mouse to start each phase of the test and to fill in the associated questionnaires (see Figure 8).



Test 2

Avez-vous compris tout de suite la devinette ?
pas du tout ○ ○ ○ ○ ○ tout à fait

Avez-vous trouvé la devinette drôle ?
pas du tout ○ ○ ○ ○ ○ tout à fait

Vous allez à présent raconter la devinette suivante à l'agent virtuel :

Quel est l'étoile la plus sale ?
C'est l'étoile d'araignée !

Vous pourrez compléter le questionnaire après avoir raconté votre devinette

Clickez sur Ok lorsque vous êtes prêt à raconter la devinette.

- chaleureux : pas du tout ○ ○ ○ ○ ○ tout à fait
- ennuyeux : pas du tout ○ ○ ○ ○ ○ tout à fait
- amusant : pas du tout ○ ○ ○ ○ ○ tout à fait
- froid : pas du tout ○ ○ ○ ○ ○ tout à fait

Figure 8: Screenshot of experiment interface.

Each participant saw four jokes in the four conditions, alternating speaking and listening conditions. Here is an example of the sequence of conditions that a participant can have: Agent speaks and smiles, Agent listens and laughs, Agent speaks and laughs, Agent listens and smiles. These sequences of condition are counter balanced to avoid any effect of their order.

Questionnaires.

To evaluate how is the act of telling a riddle perceived when the agent listens to the user’s riddle and when the agent tells a riddle to the user, we used a questionnaire similar to [14]. After watching each condition, the user had to rate two sets of factors on five degrees Likert scales:

- 3 questions: Did the participant find the riddle funny. How well s/he understood the riddle. Did s/he like the riddle.
- 6 questions related to the stance of the virtual character. Stance is defined in Scherer [28] as the “affective style that spontaneously develops or is strategically employed in the interaction with a person or a group of persons, colouring the interpersonal exchange in that situation (e.g. being polite, distant, cold, warm, supportive, contemptuous)”. We used positive qualifiers for the stance of the virtual agent: (1) Is the speaker-agent: spontaneous, warm, amusing? (2) Is the listener-agent: spontaneous, warm, amused? We used negative qualifiers: (1) Is the speaker-agent: stiff, boring, cold? (2) Is the listener-agent: stiff, bored, cold? For the stance, the questions are of the form: Do you think the agent is stiff/cold...?

Speaking agent condition.

A message pop-up on the screen explaining that the agent will tell a small joke and that the questionnaire can be filled just afterwards. When the user clicks on the “ok” button, the agent tells the joke (and smiles or laughs depending on the condition). Then the user fills in the questionnaire.

Listening agent condition.

A message pop-up on the screen, with a short riddle (two lines) and explaining that the user has to tell this story to the agent and that the questionnaire can be filled just afterwards. When the user clicks on the “ok” button, the text of the joke disappears, the user tells the story to the agent; the agent either smiles or laughs at the joke, depending on the condition. In the listening agent condition, the speech and pauses of the human participants are detected to automatically trigger the smiles and laughs of the agent at appropriate time. After having told the riddle, the user fills in the questionnaire.

5.2 Virtual agent’s behaviour and conditions

To evaluate the impact of agent’s laugh on user’s perception of the agent and of the riddle, we have considered four conditions.

- Two “test conditions” which are the laughing conditions: (1) the virtual character asks the riddle and laughs when it gives the answer; (2) the virtual character listens to the riddle and laughs when the participant gives the answer.
- Two “control conditions” which are the smiling conditions: (1) the virtual character asks the riddle and smiles when it gives the answer; (2) the virtual character listens to the riddle and smiles when the participant gives the answer.

Riddles.

Both the virtual character and the human user tell their riddle in French. When translated into English the joke is something like: “What is the future of I yawn? (speech pause) I sleep!”. According to [14] the selected four riddles are rated equivalently.

Smiles.

The smiles synthesised here correspond to the smiles validated in [14]. We used a polite smile for the question part of the riddle and an amused smile at the end of the answer.

Laughs.

The laughs that are used in the experiment are the two laughs that were described at the beginning of section 5.

5.3 Participants

Seventeen individuals participated in this study (10 female) with a mean age of 29 (SD = 5.9). They were recruited among the students and professors of our University. The participants have all spent the majority of the last five years in France and were mainly native from France (N=15). Each participant took all the four conditions. In the next section, we present in details the results of this test.

5.4 Results

To measure the effects of laughs on the user's perception, we have performed repeated measures ANOVA (each participant saw the four conditions) and the post hoc Tukey's test to evaluate the significant differences of rating between the different conditions (agent Speaks and Smiles (SS), agent speaks and laughs (SL), agent listens and smiles (LS), agent listens and laughs (LL)).

No significant differences were found between conditions for *Understanding* and *Finding funny* the riddle. No significant differences were found between conditions for the agent's *Spontaneous* and *Stiff*. Significant differences between conditions were found for the other variables: How much the agent finds the riddle funny ($F = 1.3, p < .001$), How much the agent is stiff ($F = 3.8$, $p < 0.05$), warm ($F = 6.58$, $p < .001$), boring/bored ($F = 6.23$, $p < .001$), enjoyable/amused ($F = 6.31$, $p < .001$) and cold ($F = 5.46$, $p < .001$).

The post-hoc analysis on the significant results are presented in Table 1. For each conditions pair we report results to items of the questionnaire that were given to the participants for which significant differences were found. Thus we do not report results for the various conditions presented just above (e.g. Understanding, Finding Funny the riddle). We report only the results for the qualifier *Stiff* as no significant difference has been found between *Stiff* and *Spontaneous*. In the Table 1, the first column indicates which conditions are compared (agent Speaks and Smiles (SS), agent speaks and laughs (SL), agent listens and smiles (LS), agent listens and laughs (LL)) and the first line indicates the concerned variables. The other columns are the positive and negative qualifiers for speaker-agent and listener-agent (e.g., bored /boring). The second column indicates results regarding if the agent liked the riddle (either told by the participant or by itself, depending on the condition). The inside elements of the table correspond to the condition in which the variable is significantly higher (n.s. means non significant, *: $p < .05$, **: $p < .01$, ***: $p < .001$). If in a comparison, no significant differences are found, we mark n.S.; while if there are significant differences, we indicate the condition with a higher result followed by the number of stars that gives the confidence level of the results. For instance, in Table 1, the notation LL*** at the intersection of the line LL-LS and the column Warm means that, the agent when it Listens and Laughs is perceived significantly warmer (with $p < .001$) than when it Listens and Smiles.

6. DISCUSSION

Listening conditions.

The results of the second line of Table 1 (LL-LS) tend to show that a listening agent which laughs at the joke of the user is perceived significantly more positive (warmer, more amused, less bored and less cold) than if it only smiles. When it listens, smiling agent appears to be negatively perceived (agent is considered as bored and cold).

Conditions	Agent riddle liking	Stiff /Stiff	Warm /Warm	Boring /Bored	Enjoyable /Amused	Cold
SL-SS	SL**	n.s.	n.s.	n.s.	n.s.	n.s.
LL-LS	LL***	n.s.	LL***	LS***	LL***	LS***
SS-LL	LL**	n.s.	n.s.	n.s.	LL*	n.s.
SS-LS	SS**	n.s.	n.s.	LS*	n.s.	LS*
SL-LS	SL***	LS*	SL***	LS***	SL**	LS**
SL-LL	n.s.	n.s.	n.s.	n.s.	n.s.	n.s.

Table 1: Results of ANOVA tests when comparing the pairs of conditions described in column 1 (SL vs. SS, LL vs. LS, etc). Results indicate no significant difference (n.s.), or significant difference at various levels (indicated by the number of stars). See Section 5.4 for more explanation.

Consistently with this result, participants expressed disappointment when the agent did not laugh at their joke (i.e. condition user tells a joke) and satisfaction when the agent did laugh to their joke.

Speaking conditions.

By contrast, the results of the first line of Table 1 (SL-SS) tend to show that there is not much effect of smiling vs laughing when the agent speaks: only the agent's liking of its riddle is perceived significantly higher when the agent laughs.

Smiling condition.

The results of the fourth line of Table 1 (SS-LS) show that an agent which speaks and smiles is better perceived than an agent which listens and smiles. Again the negative perception of listener-agent which "just smiles" to the user's jokes seems to explain the result.

Laughing condition.

The laughing conditions (last line of Table 1 (SL-LL)), when the agent speaks and when the agent listens, show no significant differences.

These results give a hierarchy of conditions in the context of telling a riddle:

To listen and "just smile" is the most negatively perceived attitude: the agent seems to like significantly less the joke but among others to be significantly more *bored* and *cold* than in any other condition, and to be significantly less *warm* and *amused* than in laughing conditions.

To "just smile" is perceived less negatively when the agent speaks: compared to the laughing speaking agent, only the liking of the riddle is lower.

Laughing does not appear to change the perception when the agent speaks or listens whereas smiling does: "just" smiling when listening is perceived negatively.

The laugh synthesised animation clearly enriched the agent with fine interaction capacities, and our study points out that this laugh contrasts with smiles through two facets: (1) when laugh is triggered in reaction to the partner's talk, it appears as a reward and a very interactive behaviour; (2) when laugh is triggered by the speaker itself, it appears as more self-centred behaviour, an epistemic stance.

7. CONCLUDING COMMENTS

We presented a laughter motion synthesis model that takes as input pseudo-phonemes and their duration as well as speech features to compute a synchronized multimodal animation. We evaluated our model to check how laughing agent is perceived when telling / listening to a joke.

Contrasting with one of our expectations, we did not found any effect of agent's laugh on human user's liking of the joke. This may be explained by the fact that human had to read the joke before telling it to the agent: thus they had already evaluated the joke while reading it for themselves before telling it to the agent and seeing its reaction.

However, our data shows that laugh induces a significant positive effect in the context of telling a riddle, when the agent is listening and reacting to the user. The effect is less clear when the agent is speaking, certainly due to this very context of telling a riddle: laughing at its own joke is more an epistemic stance (concerning what the speaker thinks of what it says) than a social stance (i.e. a social attitude directed toward the partner).

8. ACKNOWLEDGMENTS

The research leading to these results has received partial funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n. 270780, ILHAIRE project. We are very grateful to Cereproc for letting us use their voice synthesizer (www.cereproc.com).

9. REFERENCES

[1] V. Adelsward. Laughter and dialogue: The social significance of laughter in institutional discourse. *Nordic Journal of Linguistics*, 102(12):107–136, 1989.

[2] P. Boersma and D. Weeninck. Praat, a system for doing phonetics by computer. *Glot International*, 5(9/10):341–345, 2001.

[3] C. Busso, Z. Deng, U. Neumann, and S. Narayanan. Natural head motion synthesis driven by acoustic prosodic features. *Journal of Visualization and Computer Animation*, 16(3-4):283–290, 2005.

[4] 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*, pages 21–24, 2009.

[5] P. C. DiLorenzo, V. B. Zordan, and B. L. Sanders. Laughing out loud: control for modeling anatomically inspired laughter using audio. *ACM Trans. Graph.*, 27(5):125, 2008.

[6] Y. Ding, C. Pelachaud, and T. Artières. Modeling multimodal behaviors from speech prosody. In *IVA*, pages 217–228. 2013.

[7] Y. Ding, M. Radenen, T. Artières, and C. Pelachaud. Speech-driven eyebrow motion synthesis with contextual markovian models. In *ICASSP*, pages 3756–3760, 2013.

[8] S. Mariooryad and C. Busso. Generating human-like behaviors using joint, speech-driven models for conversational agents. *IEEE Trans. on Audio, Speech & Language Processing*, 20(8):2329–2340, 2012.

[9] R. Martin. Is laughter the best medicine? humor, laughter, and physical health. *Current Directions in Psychological Science*, 11(6):216–220, 2002.

[10] M. Neff and E. Fiume. Modeling tension and relaxation for computer animation. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '02.

[11] 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 *AAMAS*, pages 619–626, 2013.

[12] R. Niewiadomski and C. Pelachaud. Towards multimodal expression of laughter. In *IVA*, pages 231–244, 2012.

[13] R. Niewiadomski, J. Urbain, C. Pelachaud, and T. Dutoit. Finding out the audio and visual features that influence the perception of laughter intensity and differ in inhalation and exhalation phases. In *International Workshop on Corpora for Research on EMOTION SENTIMENT and SOCIAL SIGNALS, LREC 2012*.

[14] M. Ochs and C. Pelachaud. Model of the perception of smiling virtual character. In *AAMAS*, pages 87–94, 2012.

[15] I. Pandzic and R. Forcheimer. *MPEG4 Facial Animation - The standard, implementations and applications*. John Wiley & Sons, 2002.

[16] K. Perlin. Improving noise. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '02, pages 681–682.

[17] R. Provine. Laughter. *American Scientist*, 84(1):38–47, 1996.

[18] W. Ruch and P. Ekman. The Expressive Pattern of Laughter. *Emotion qualia, and consciousness*, pages 426–443, 2001.

[19] J. Saragih, S. Lucey, and J. Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2):200–215, 2011.

[20] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura. Speech parameter generation algorithms for hmm-based speech synthesis. In *ICASSP*, pages 1315–1318, 2000.

[21] J. Urbain, E. Bevacqua, T. Dutoit, A. Moinet, R. Niewiadomski, C. Pelachaud, B. Picart, J. Tilmanne, and J. Wagner. The avlaughtercycle database. In *LREC*, 2010.

[22] J. Urbain, H. Çakmak, and T. Dutoit. Automatic phonetic transcription of laughter and its application to laughter synthesis. In *biannual Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 153–158, 2013.