Real-Time Prosody-Driven Synthesis of Body Language

Sergey Levine Christian Theobalt Vladlen Koltun

Stanford University *

Figure 1: *Data-driven body language is synthesized from live speech input.*

Abstract

Human communication involves not only speech, but also a wide variety of gestures and body motions. Interactions in virtual environments often lack this multi-modal aspect of communication. We present a method for automatically synthesizing body language animations directly from the participants' speech signals, without the need for additional input. Our system generates appropriate body language animations by selecting segments from motion capture data of real people in conversation. The synthesis can be performed progressively, with no advance knowledge of the utterance, making the system suitable for animating characters from live human speech. The selection is driven by a hidden Markov model and uses prosody-based features extracted from speech. The training phase is fully automatic and does not require hand-labeling of input data, and the synthesis phase is efficient enough to run in real time on live microphone input. User studies confirm that our method is able to produce realistic and compelling body language.

CR Categories: I.3.6 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: Human Animation, Data-Driven Animation, Control, Nonverbal Behavior Generation, Gesture Synthesis

1 Introduction

Interactions between human characters are often the most interesting aspects of networked virtual environments. Modern real-time graphics technology can endow these characters with a photo-realistic appearance, but is still unable to generate the vast variety of motions that real human beings exhibit. Gestures and speech coexist in time and are tightly intertwined [McNeill 1992], but current input devices are far too cumbersome to allow body language to be conveyed as intuitively and seamlessly as it would be in person. Current virtual worlds frequently employ keyboard or mouse commands to allow participants to utilize a small library of pre-recorded gestures, but this mode of communication is unnatural for extemporaneous body language. Given these limitations on direct input, body language for human characters must be synthesized automatically in order to produce consistent and believable results.

We present a data-driven method that automatically generates body language animation from the prosody of the participant's speech signal. The system is trained on motion capture data of real people in conversation, with simultaneously recorded audio. Our main contribution is a method for modeling the gesture formation process that is appropriate for progressive real-time synthesis, as well as an efficient algorithm that uses this model to produce an animation from a live speech signal, such as a microphone.

To generate the animation, we select appropriate gesture subunits from the motion capture training data based on prosody cues in the speech signal. Prosody cues are known to correspond well to emotional state [Adolphs 2002; Schröder 2004] and emphasis [Terken 1991]. Gesture has also been observed to reflect emotional state [Wallbot 1998; Montepare et al. 1999] and highlight emphasized phrases [McNeill 1992]. The selection is performed by a specialized hidden Markov model (HMM), which ensures that the gesture subunits transition smoothly and are appropriate for the tone of the current utterance. In order to synthesize gestures in real time, the HMM predicts the next gesture and corrects mispredictions. By using coherent gesture subunits with appropriate transitions enforced by the HMM, we ensure a smooth and realistic animation. The use of prosody for driving the selection of motions also ensures that the synthesized animation matches the timing and tone of the speech.

We also present four user studies that compare synthesized animations for novel utterances with the original motion capture sequences corresponding to each utterance. The results of the studies, presented in Section 10, confirm that our method produces compelling body language and generalizes to different speakers.

By animating human characters in real time with no additional input, our system can seamlessly produce plausible body language for human-controlled characters, thus improving the immersiveness of interactive virtual environments and the fluidity of virtual conversations. To the best of our knowledge, this is the first proposed audio-driven method for body language synthesis that can generate animations from live speech.

*e-mail:{svlevine,theobalt,vladlen}@cs.stanford.edu

2 Related Work

Although no system has been proposed that both synthesizes full-body gestures and operates in real time on live voice input, a number of methods have been devised that synthesize either full-body or facial animations from a variety of inputs. Such methods often aim to animate Embodied Conversational Agents (ECAs), which operate on a pre-defined script or behavior tree [Cassell 2000], and therefore allow for concurrent planning of synthesized speech and gesture to ensure co-occurrence. Often these methods rely on the content author to specify gestures as part of the input, using a concise annotation scheme [Hartmann et al. 2002; Kopp and Wachsmuth 2004]. New, more complete annotation schemes for gestures are still being proposed [Kipp et al. 2007], and there is no clear consensus on how gestures should be specified. Some higher-level methods also combine behavioral planning with gesture and speech synthesis [Cassell et al. 1994; Perlin and Goldberg 1996], with gestures specified as part of scripted behavior. However, all of these methods rely on an annotation scheme that concisely and completely specifies the desired gestures. Stone et al. [2004] avoid the need for annotation of the input text with a data-driven method, which re-arranges pre-recorded motion capture data to form the desired utterance. However, this method is limited to synthesizing utterances made up of pre-recorded phrases, and does require the hand-annotation of all training data. We also employ a data-driven method, but our system is able to automatically retarget appropriate pre-recorded motion to an arbitrary utterance.

Several methods have been proposed that operate on arbitrary input, such as text. Cassell et al. [2001] propose an automatic rule-based gesture generation system for ECAs using natural language processing, while Neff et al. [2008] use a probabilistic synthesis method trained on hand-annotated video. However, both of these methods rely on concurrent generation of speech and gesture from text. Text does not capture the emotional dimension that is so important to body language, and neither text communication, nor speech synthesized from text can produce as strong an impact as real conversation [Jensen et al. 2000].

Animation directly from voice has been explored for synthesizing facial expressions and lip movements, generally as a data-driven approach using some form of probabilistic model. Bregler et al. [1997] propose a video-based method that reorders frames in a video sequence to correspond to a stream of phonemes extracted from speech. This method is further extended by Brand [1999] by retargeting the animation onto a new model and adopting a sophisticated hidden Markov model for synthesis. Hidden Markov models are now commonly used to model the relationship between speech and facial expression [Li and Shum 2006; Xue et al. 2006]. Other automatic methods have proposed synthesis of facial expressions using more sophisticated morphing of video sequences [Ezzat et al. 2002], physical simulation of muscles [Sifakis et al. 2006], or by using hybrid rule-based and data-driven methods [Beskow 2003].

Although speech-based synthesis of facial expressions is quite common, it does not generally utilize vocal prosody. Since facial expressions are dominated by mouth movement, many speech-based systems use techniques similar to phoneme extraction to select appropriate mouth shapes. However, a number of methods have been proposed that use speech prosody to model expressive human motion beyond lip movements. Albrecht et al. [2002] use prosody features to drive a rule-based facial expression animation system, while more recent systems apply a data-driven approach to generate head motion from pitch [Chuang and Bregler 2005] and facial expressions from vocal intensity [Ju and Lee 2008]. Incorporating a more sophisticated model, Sargin et al. [2007] use prosody features to directly drive head orientation with a HMM. Although these methods only animate head orientation from prosody, Morency et al. [2007] suggest that prosody may also be useful for predicting gestural displays.

Our system selects appropriate motions using a prosody-driven HMM reminiscent of the above techniques, but with each output segment corresponding to an entire gesture subunit, such as a "stroke" or a "hold." This effectively reassembles the training motion segments into a new animation. Current techniques for assembling motion capture into new animations generally utilize some form of graph traversal to select the most appropriate transitions [Arikan and Forsyth 2002; Kovar et al. 2002; Lee et al. 2002]. Our system captures a similar notion in the structure of the HMM, with high-probability hidden-state transitions corresponding to transitions that occur frequently in the training data. While facial animation synthesis HMMs have previously used remapping of observations [Brand 1999], or conditioned the output animations on the input audio [Li and Shum 2006], we map animation states directly to the hidden states of our model. This allows us to design a simpler system that is able to deal with coherent gesture subunits without requiring them to directly coincide in time with audio segments.

While the methods described above are able to synthesize plausible body language or facial expression for ECAs, none of them can generate full-body animations from live speech. Animating human-controlled characters requires real-time speeds and a predictive model that handles arbitrary speech without the need for manual annotation or knowledge of the entire utterance. Such a model constitutes the main contribution of this work.

3 Background on Gesture and Speech

The most widely-used taxonomy for gestures was proposed by McNeill [1992], though he later suggested that a more continuous classification would be more appropriate [2005]. McNeill's original taxonomy consists of four gesture types: iconics, metaphorics, deictics, and beats. Iconics present images of concrete objects or actions, metaphorics represent abstract ideas, deictics serve to locate entities in space, and beats are simple, repetitive motions meant to emphasize key phrases [McNeill 1992].

McNeill noted that "beats tend to have the same form regardless of content," and that they are generally used to highlight emphasized words. He also observed that beats constitute half of all gestures, and nearly two-thirds of gestures accompanying non-narrative speech [McNeill 1992]. Since prosody correlates well to emphasis [Terken 1991], it should correspond to the emphasized words that beats highlight, making it particularly useful for synthesizing this type of gesture.

In addition, there is evidence that prosody carries much of the emotive content of speech [Adolphs 2002; Schröder 2004], and that emotional state is often reflected in body language [Wallbot 1998; Montepare et al. 1999]. Therefore, we expect a prosody-driven system to produce accurately timed and appropriate beats, as well as more complex abstract or emotion-related gestures with the appropriate emotive content. The accompanying video presents examples of synthesized animations for emphasized and emotional utterances.

We capture prosody using the three features that it is most commonly associated with: pitch, intensity, and duration. Pitch and intensity have previously been used to drive expressive faces [Chuang and Bregler 2005; Ju and Lee 2008], duration has a natural correspondence to the rate of speech, and all of these aspects of prosody are informative for determining emotional state [Scherer et al. 1991]. While semantic meaning also corresponds strongly to gesture, we do not attempt to interpret the utterance. This approach has some limitations, as discussed in Section 11, but is more appropriate for online, real-time synthesis. Extracting semantic meaning from speech to synthesize gesture without knowledge of the full utterance is difficult because it requires a *predictive* speech interpreter, while prosody can be obtained efficiently without looking ahead in the utterance.

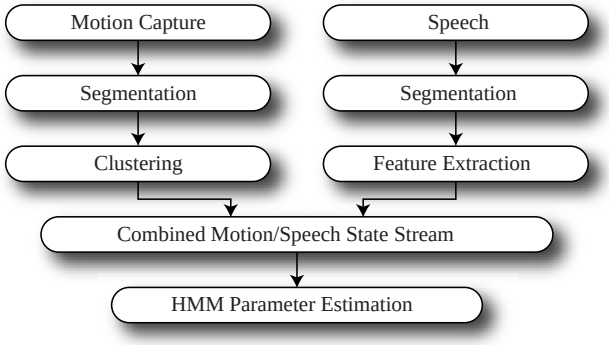
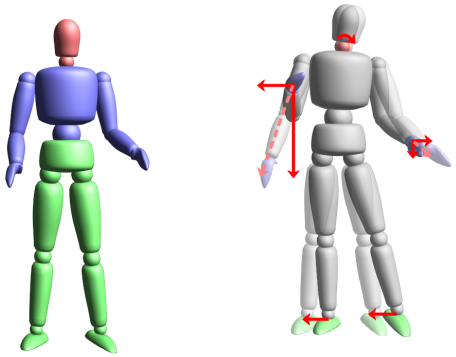


Figure 2: *Training of the body language model from motion capture data with simultaneous audio.*



(a) The three sections. (b) Displacement of key parts.

Figure 3: *For each of the body sections in (a), a set of body parts is used to summarize segment dynamics with average and maximum displacement from the first frame (b), as well as average velocity. The head section uses rotation of the neck, the arms use hand positions, and the lower body uses the positions of the feet.*

4 Overview

During the training phase, our system processes a corpus of motion capture and audio data to build a probabilistic model that correlates gesture and prosody. The motion capture data is processed by extracting gesture subunits, such as "strokes" and "holds." These subunits are then clustered to identify recurrences of the same motion, as described in Section 5. The training speech signal is processed by extracting syllables and computing a set of prosody-based features on each syllable, as described in Section 6. Finally, the parameters of a hidden Markov model, which is described in Section 7, are estimated directly from the two resulting state-streams. This process is summarized in Figure 2. The HMM can then be used to automatically synthesize appropriate body language for a live novel utterance in real time, using the algorithm described in Section 8.

The correlation between speech and body language is inherently a many-to-many mapping: there is no unique animation that is most appropriate for a given utterance [Brand 1999]. This ambiguity makes validation of synthesis techniques difficult. Since the only way to judge the quality of a synthesized body language animation is by subjective evaluation, we conducted a survey to validate our method. We present the results of this study in Section 10.

5 Motion Data Processing

The final training set for our system consists of 30 minutes of motion capture data with accompanying speech, divided into six scenes. An additional 15 minute training set from a different speaker was also used in the user study. Each scene in each set was excerpted from an extemporaneous conversation, ensuring that the body language was representative of real human interaction. Typical conversation topics were travel, movies, and politics. The actors were asked to avoid prominent iconic gestures that might depend strongly on semantics, but no other instructions were given.

The motion capture data is mapped onto a 14-joint skeleton and segmented into gesture unit phases, henceforth referred to as gesture subunits. These segments are then clustered to identify recurrences of similar motions.

5.1 Motion Segmentation

Current motion segmentation methods identify broad categories of motion within a corpus of motion capture data, such as walking and sitting [Barbič et al. 2004; Müller et al. 2005], while much of the existing work on data-driven gesture animation segments training data manually [Stone et al. 2004; Neff et al. 2008]. Perceptually distinct gesture subunits are not as dissimilar as these broad categories, while manual annotation does not scale gracefully to large amounts of data. Therefore, we propose a new method for segmentation inspired by the current understanding of gesture structure.

Gestures consist of pre-stroke hold, stroke, post-stroke hold, and retraction phases [McNeill 1992; Kendon 2004]. Such phases have previously been used to segment hand gestures [Majkowska et al. 2006]. From this we deduce that a gesture unit consists of alternating periods of fast and slow motion. Therefore, we place segment boundaries when we detect a shift from slow motion into fast motion, or vice versa, using an algorithm inspired by Fod et al. [2002]. For each frame, we compute $z = \sum_{j=1}^{14} u_j ||\omega_j||^2$, the weighted sum of the squared magnitudes of the angular velocities of the joints, denoted by ω_j for joint j. The weights u_j correspond to the estimated perceptual importance of joints, with high-influence joints such as the pelvis, abdomen, and hips weighted higher, and low-influence joints such as the hands weighted lower. To avoid creating small segments due to noise, we only create segments that exceed either a minimum length or a minimum limit on the integral of z over the segment. To avoid unnaturally long still segments during long pauses, we place an upper bound on segment length. The motions of the head, arms, and lower body do not always coincide, so we segment these three body sections separately and henceforth treat them as separate animation streams. The joints that constitute each of the sections are illustrated in Figure 3. Our final training set contains 2542 segments for the head, 2388 for the arms, and 1799 for the lower body.

5.2 Segment Clustering

Once the motion data has been segmented, we cluster the segments to identify recurring gesture subunits. Our clustering algorithm is based on Ward hierarchical clustering [Ward 1963], though a variety of methods may be appropriate. Segments are compared according to a three-part distance function that takes into account length, starting pose, and a fixed-size "summary" of the dynamics of the segment. The "summary" holds some measure of velocity and maximum and average displacement along each of the axes for a few key body parts in each section, as in Figure 3. The head section uses the rotation of the neck about each of the axes. The arm section uses the positions of the hands, which often determine the perceptual similarity of two gestures. The lower body uses the positions of the feet relative to the pelvis, which capture motions such as the shifting of weight. For the arms and lower body, velocity and maximum and average displacement each constitute a six-dimensional vector (three axes per joint), while for the head this vector has three dimensions. Each vector is rescaled to a logarithmic scale, since larger gestures can tolerate greater variation while remaining perceptually similar. Using six-dimensional vectors serves to de-emphasize the unused hand in one-handed gestures: if instead a three-dimensional

vector for each hand was rescaled, small motions in the unused hand would be weighted too heavily on the logarithmic scale relative to the more important large motion of the active hand. To compare two summaries, we use the sum of the distances between the three vectors. The full training set was clustered into 20 gesture subunits for the head, 45 for the arms, and 6 for the lower body.

6 Audio Processing

To obtain prosody features, we continuously extract pitch and intensity curves from the audio stream. After segmentation, these curves are used to compute a concise prosody descriptor for each audio segment, describing the inflection, intensity, and duration of the syllable. Pitch is extracted using the autocorrelation method, as described in [Boersma 1993], and intensity is extracted by squaring waveform values and filtering them with a Gaussian analysis window. For both tasks, we use components of the Praat speech analysis tool [Boersma 2001].

Gesture strokes have been observed to consistently end at or before, but never after, the prosodic stress peak of the accompanying syllable. This is referred to as one of the gesture "synchrony rules" by McNeill [1992]. Therefore, we segment the audio signal by syllables, and compute feature values on each syllable. For efficient segmentation, we used a simple algorithm inspired by Maeran et al. [1997], which identifies peaks separated by valleys in the intensity curve, under the assumption that distinct syllables will have distinct intensity peaks. In order to segment the signal progressively in real time, we identify a new syllable as soon as the potential peak of the *next* syllable is discovered. Because of this, syllable observations are issued at or before the peak of the next syllable, which allows us to more easily follow the synchrony rule.

In order to train the system on a small corpus of training data, we limit the size of the observation state space by using a small set of discrete features, rather than the more standard mixture of Gaussians approach. As described in Section 3, we utilize pitch, intensity, and the duration of syllables. Pitch is described using two binary features indicating the presence or absence of significant upward or downward inflection, corresponding to one standard deviation above the mean. Intensity is described using a trinary feature indicating silence, standard intensity, or high intensity, corresponding to half a standard deviation above the mean. Length is also described using a trinary feature, with "long" segments one deviation above the mean and "short" segments half a deviation below. The means and deviations of the training sequence are established prior to feature extraction. During synthesis, the means and deviations are estimated incrementally from the available audio data as additional syllables are observed.

7 Probabilistic Gesture Model

Previously proposed HMM-driven animation methods generally use temporally aligned animation and speech segments. The input audio is remapped directly to animation output [Brand 1999], or is coupled to the output in a more complex manner, such as conditioning output and transition probabilities on the input state [Li and Shum 2006]. The HMM itself is trained with some variation of the EM algorithm [Li and Shum 2006; Xue et al. 2006].

Since our input and output states are not temporally aligned, and since each of our animation segments corresponds to a meaningful unit, such as a "stroke" or "hold," we take a different approach and map animation segments directly to the hidden states of our model, rather than inferring hidden structure with the EM algorithm. This allows us greater control in designing a specialized system for dealing with the lack of temporal alignment even when using a small amount of training data, though at the expense of being unable to infer additional hidden structure beyond that provided by the clustering of gesture subunits.

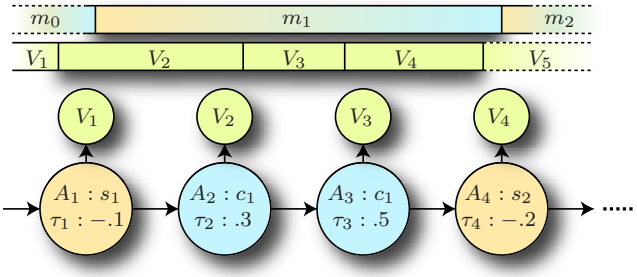


Figure 4: *The hidden state space of our model consists of the index of the active motion, $A_t = s_i$ or c_i, and the fraction of that motion which has elapsed up to this point, τ_t. Observations V_t are paired with the active motion at the end of the next syllable.*

7.1 Gesture Model State Space

The process of gesture subunit formation consists of the selection of appropriate motions that correspond well to speech and line up in time to form a continuous animation stream. As discussed in Section 6, gesture strokes terminate at or before syllable peaks, and syllable observations arrive when the peak of the next syllable is encountered. In our training data, less than 0.4% of motion segments did not contain a syllable observation, so we may assume that at least one syllable will be observed in every motion, though we will often observe more. Therefore, syllable observations provide a natural discretization for continuous gesture subunit formation, allowing us to model this process with a discrete-time HMM.

Under this discretization, we assume that gesture formation is a Markov process in which H_t, the animation state at syllable observation V_t, depends only on V_t and the previous animation state H_{t-1}. As discussed in Section 6, V_t contains a small set of discrete prosody features, and may be expressed as the index of a discrete observation state, which we will denote v_j. We define the animation state $H_t = \{A_t, \tau_t(i)\}$, where $A_t = m_i$ is the index of the motion cluster corresponding to the desired motion, and $\tau_t(i)$ is the fraction of that motion which has elapsed up to the observation V_t. This allows motions to span multiple syllables.

To prevent adjacent motions from overlapping, we could use a temporal scaling factor to ensure that the current motion terminates precisely when the next motion begins. However, during online synthesis, we have no knowledge of the length of future syllables, and therefore would not be able to accurately predict the scaling factor. Instead, we interrupt a motion and begin a new one when it becomes unlikely that another syllable will be observed within that motion, thus ensuring that motions terminate at syllable observations and do not overlap. While this reduces the quality of the animation, it allows the gestures strokes to be synchronized to syllable peaks without time warping. In practice, segments were interrupted on average 77% of the way to completion on 10 minutes of novel utterances from typical conversations, indicating that most segments run almost to completion, and the discontinuity resulting from the synchronizing interruption is minor.

7.2 Parameterizing the Gesture Model

In order to compactly express the parameters of the gesture model, we first note that τ evolves in a very structured manner. If observation V_t corresponds to the continuation of the current motion m_i, then $A_t = A_{t-1} = m_i$ and $\tau_t(i) = \tau_{t-1}(i) + \triangle\tau_t(i)$, where $\triangle\tau_t(i)$ is the fraction of the motion m_i which has elapsed since the previous observation. If instead the current motion terminates at observation V_t and is succeeded by $A_t = m_j$, then $\tau_t(j) = 0$. Let $\tau'_t(i) = \tau_{t-1}(i) + \triangle\tau_t(i)$, then τ_t is completely determined by τ'_t and A_t, except in the case when motion m_i follows itself, and $A_t = A_{t-1} = m_i$ but $\tau_t(i) = 0$. To disambiguate this case, we introduce a start state s_i and a continuation state c_i for each motion

cluster m_i, so that A_t may take on either s_i or c_i (we will denote an arbitrary animation state as a_j). This approach is similar to the "BIO notation" (beginning, inside, outside) used in semantic role labeling [Ramshaw and Marcus 1995]. Under this new formulation, $\tau_t(i) = \tau'_t(i)$ when $A_t = c_i$ and $A_{t-1} = c_i$ or s_i. Otherwise, $\tau_t(i) = 0$. Therefore, τ_t is completely determined by τ'_t and A_t.

The transition probabilities for A_t are a function of $A_{t-1} = m_i$ and $\tau'_t(i)$. While this distribution may be learned from the training data, we can simplify it considerably. Although the precise length of a motion may vary slightly for syllable alignment, we assume that the transitions out of that motion do not depend this length. Therefore, we can define T_{c_i} as the vector of transition probabilities out of a motion m_i:

$$T_{c_i,s_j} = P(A_t = s_j | A_{t-1} = c_i, A_t \neq c_i).$$

We can also assume that the continuation of m_i depends only on $\tau'_t(i)$, since, if $A_t = c_i$, $A_{t-1} = c_i$ or s_i, providing little additional information. Intuitively, this seems reasonable, since encountering another observation within a motion becomes less probable as the motion progresses. Therefore, we can construct the probabilities of transitions from c_i as a linear combination of T_{c_i} and e_{c_i}, the guaranteed transition into c_i, according to some function f_i of $\tau'_t(i)$:

$$P(A_t | A_{t-1} = c_i) = \begin{cases} f_i(\tau'_t(i)) & A_t = c_i \\ (1 - f_i(\tau'_t(i)))T_{c_i,A_t} & \text{otherwise} \end{cases} \quad (1)$$

We now define the vector T_{s_i} as the distribution of transitions out of s_i, and construct the full transition matrix $T = [T_{s_1}, T_{c_1}, ..., T_{s_n}, T_{c_n}]$, where n is the total number of motion clusters. The parameters of the HMM are then completely specified by the matrix T, a matrix of observation probabilities O given by $O_{i,j} = P(v_i|a_j)$, and the interpolation functions $f_1, f_2, ..., f_n$.

7.3 Estimating Model Parameters

We estimate the matrices directly from the training data by counting the frequency of transitions and observations. When pairing observations with animation states, we must consider the animation state that is active at the end of the *next* observation, as shown in Figure 4. When a motion m_i terminates on the syllable V_t, we must predict the next animation segment. However, if we associate V_t with the *current* segment, the observed evidence would indicate that c_i is the most probable state, which is not the case. Instead, we would like to predict the *next* state, which is precisely the state that is active at the end of the next syllable V_{t+1}.

The value of an interpolation function, $f_i(\tau_t(i))$, gives the probability that another syllable will terminate within motion m_i after the current one, which terminated at point $\tau_t(i)$. Since motions are more likely to terminate as $\tau_t(i)$ increases, we may assume that f_i is monotonically decreasing. From empirical observation of the distribution of τ values for the final syllables in each motion segment, we concluded that a logistic curve would be well suited for approximating f_i. Therefore, the interpolation functions are estimated by fitting a logistic curve to the τ values for final syllables within each training motion cluster by means of gradient descent.

8 Synthesis

The model assembled in the training phase allows our system to animate a character in real time from a novel live speech stream. Since the model is trained on animation segments that *follow* audio segments, it is able to predict the most appropriate animation as soon as a new syllable is detected. As noted earlier, gesture strokes terminate at or before the intensity peak of the accompanying syllable [McNeill 1992]. The syllable segmentation algorithm detects a syllable in continuous speech when it encounters the syllable peak of the *next* syllable, so new gestures begin at syllable peaks. Consequently, the previous gesture ends at or before a syllable peak.

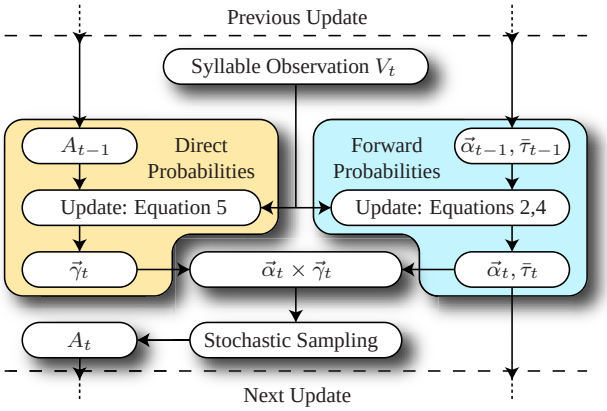


Figure 5: *Diagram of the synthesis algorithm for one observation. A_t is the t^{th} animation state, V_t is the t^{th} observation, $\vec{\alpha}_t$ is the forward probability vector, $\bar{\tau}_t$ is the vector of expected τ values, and $\vec{\gamma}_t$ is the direct probability vector.*

To synthesize the most probable and coherent animation, we compute a vector of forward probabilities along with direct probabilities based on the previously selected hidden state. The forward probabilities carry context from previous observations, while the direct probabilities indicate likely transitions given only the last displayed animation and current observation. Together, these two vectors may be combined to obtain a prediction that is both *probable* given the speech context and *coherent* given the previously displayed state, resulting in animation that is both smooth and plausible. The synthesis process is illustrated in Figure 5.

8.1 Forward Probabilities

Given the parameters of a hidden Markov model and a sequence of syllable-observations $\{V_1, V_2, ..., V_t\}$, the most probable hidden state at time t can be efficiently determined from forward probabilities [Rabiner 1989]. Since transition probabilities depend on the value $\tau_t(i)$ at the current observation, we must take it into account when updating forward probabilities. To this end, we maintain both a vector of forward probabilities $\vec{\alpha}_t$, and a vector $\bar{\tau}_t$ of the expected values $E(\tau_t(i)|A_t = c_i)$ at the t^{th} observation.

As before, we define a vector $\bar{\tau}'_t(i) = \bar{\tau}_{t-1}(i) + \triangle\tau_t(i)$. Using this vector, computing the transition probabilities between any two states at observation V_t is analogous to Equation 1:

$$P(A_t | A_{t-1}) = \begin{cases} f_i(\bar{\tau}'_t(i)) & A_t = A_{t-1} = c_i \\ (1 - f_j(\bar{\tau}'_t(j)))T_{c_j,A_t} & A_{t-1} = c_j \\ T_{A_{t-1},A_t} & \text{otherwise} \end{cases}$$

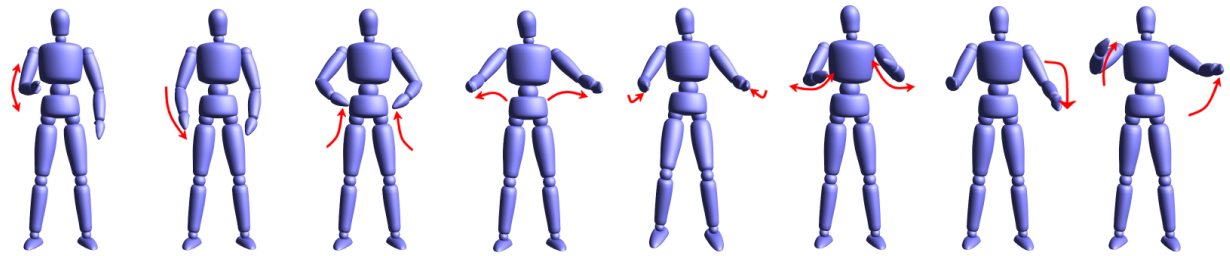
With this formula for the transition between any two hidden states, we can use the usual update rule for $\vec{\alpha}_t$:

$$\alpha_t(i) = \eta \sum_{k=1}^{2n} \alpha_{t-1}(k) P(A_t = a_i | A_{t-1} = a_k) P(V_t | a_i), \quad (2)$$

where η is the normalization value. Once the forward probabilities are computed, $\bar{\tau}_t$ must also be updated. This is done according to the following update rule:

$$\bar{\tau}_t(i) = E(\tau_t(i)|A_t = c_i) =$$

$$\sum_{k=1}^{2n} \frac{P(A_t = c_i, A_{t-1} = a_k) E(\tau_t(i)|A_t = c_i, A_{t-1} = a_k)}{P(A_t = c_i)}. \quad (3)$$



"Motorcycles become kind of a fully immersive experience, where the sound of it, the vibration, the seating position, it all matters."

Figure 6: *An excerpt from a synthesized animation.*

Since only s_i and c_i can transition into c_i, $P(A_t = c_i, A_{t-1}) = 0$ if $A_{t-1} \neq c_i$ and $A_{t-1} \neq s_i$. If $A_{t-1} = s_i$, this is the first event during this motion, so the previous value $\tau_{t-1}(i)$ must have been zero. Therefore, $E(\tau_t(i)|A_t = c_i, A_{t-1} = s_i) = \triangle\tau_t(i)$. If $A_{t-1} = c_i$, the previous expected value $\tau_{t-1}(i)$ is simply $\bar{\tau}_{t-1}(i)$, so the new value is $E(\tau_t(i)|A_t = c_i, A_{t-1} = c_i) = \bar{\tau}_{t-1}(i) + \triangle\tau_t(i) = \bar{\tau}'_t(i)$. Therefore, we can reduce Equation 3 to:

$$\begin{aligned}\bar{\tau}_t(i) &= \frac{P(A_t = c_i, A_{t-1} = c_i)E(\tau_t(i)|A_t = c_i, A_{t-1} = c_i)}{P(A_t = c_i)} \\ &+ \frac{P(A_t = c_i, A_{t-1} = s_i)E(\tau_t(i)|A_t = c_i, A_{t-1} = s_i)}{P(A_t = c_i)} \\ &= \frac{\alpha_{t-1}(c_i)f_i(\bar{\tau}'_t(i))\bar{\tau}'_t(i) + \alpha_{t-1}(s_i)T_{s_i,c_i}\triangle\tau_t(i)}{\alpha_t(c_i)}. \quad (4)\end{aligned}$$

Once we compute the forward probabilities and $\bar{\tau}_t$, we could obtain the most probable state from $\alpha_t(i)$. However, choosing the next state based solely on forward probabilities would produce an erratic animation stream, since the most probable state at a given observation need not follow smoothly from the most probable state at the previous observation.

8.2 Most Probable and Coherent State

Since we have already displayed the previous animation state A_{t-1}, we can estimate the current state from only the previous state and current observation to ensure a high-probability transition, and thus a coherent animation stream. Given $\tau'_t(i) = \tau_{t-1}(i) + \triangle\tau_t(i)$ for currently active motion i, we compute transition probabilities just as we did in Section 7.2. If the previous state is the start state s_i, transition probabilities are given directly as $P(A_t|A_{t-1} = s_i) = T_{s_i,A_t}$. If the previous state is the continuation state c_i, the transition probability $P(A_t|A_{t-1} = c_i)$ is given by Equation 1. With these transition probabilities, we compute the vector of direct probabilities $\vec{\gamma}_t$ in the natural way as:

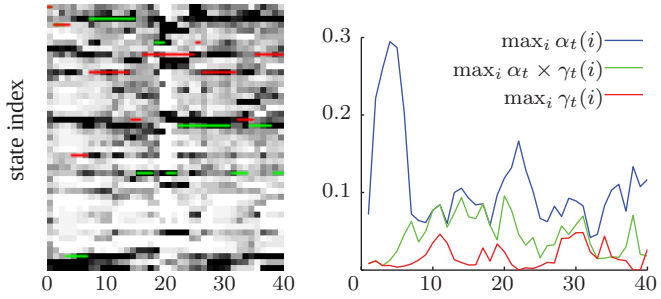


Figure 7: *The image shows the states selected over* 40 *observations with* $\max_i \gamma_t(i)$ *(red) and* $\max_i \alpha_t(i) \times \gamma_t(i)$ *(green), with dark cells corresponding to high forward probability. The graph plots the exact probability of the selections, as well as* $\max_i \alpha_t(i)$.

$$\gamma_t(i) = \eta P(A_t = a_i|A_{t-1})P(V_t|a_i). \quad (5)$$

Using $\vec{\gamma}_t$ directly, however, would quickly drive the synthesized animation down an improbable path, since context is not carried forward. Instead, we use $\vec{\gamma}_t$ and $\vec{\alpha}_t$ together to select a state that is both probable given the sequence of observations and coherent given the previously displayed animation state. We compute the final distribution for the current state as the normalized pointwise product of the two distributions $\vec{\alpha}_t \times \vec{\gamma}_t$, which preserves the coherence of $\vec{\gamma}_t$. As shown in Figure 7, this method also generally selects states that have a higher forward probability than simply using $\vec{\gamma}_t$ directly. The green lines, representing states selected using the combined method, tend to coincide with darker areas, representing higher forward probabilities, though they deviate as necessary to preserve coherence. Note that this method is heuristic, see [Treuille et al. 2007] for a principled approach to a related problem.

To obtain the actual estimate, we could simply select the most probable state, as we did in Figure 7. However, always displaying the "optimal" animation is not necessary, since various gestures are often appropriate for the same speech segment. We instead select the current state by stochastically sampling the state space according to this product distribution. This has several desirable properties: it introduces greater variety into the animation, prevents grating repetitive motions, and makes the algorithm less sensitive to issues arising from the specific choice of clusters.

8.3 Early and Late Termination

As discussed in Section 7.1, we account for variation in motion length by terminating a motion if we are unlikely to encounter another syllable within that motion. In the case that we mispredict this event and do not terminate a motion, the motion may end between syllable observations. To handle this case, we re-examine the last observation by restoring $\vec{\alpha}_{t-1}$ and $\bar{\tau}_{t-1}$ and performing the update again, with the new evidence that, if $A_{t-1} = c_i$ or s_i, $A_t \neq c_i$. This corrects the previous "misprediction," though the newly selected motion is not always as far along as it should be, since it must start from the beginning.

9 Constructing the Animation Stream

In the previous section, we discussed how an animation state is selected. Once the animation state and its corresponding cluster of animation segments is chosen, we must blend an appropriate animation segment from this cluster into the animation stream. We make the actual segment selection by considering only those segments in the cluster that begin in a pose which is within some tolerance to that of the last frame. The tolerance is a constant factor of the pose difference to the nearest segment in the cluster, ensuring that at least one segment is always available. One of the segments that fall within this tolerance is randomly selected. Random selection avoids jarring repetitive motion when the same animation state occurs multiple times consecutively.

Once the appropriate segment is selected, it must be blended with the previous frame to create a smooth transition. Although linear blending is generally sufficient for a perceptually smooth transition [Wang and Bodenheimer 2008], it requires each motion to have enough extra frames at the end to accommodate a gradual blend, and simply taking these frames from the original motion capture data may introduce extra, unwanted gestures. For many of our motions, the blend interval would also exceed the length of the gesture, resulting in a complex blend between many segments.

Instead, we use a velocity-based blending algorithm, which keeps the magnitude of the angular velocity on each joint equal to that of the desired animation, and adjusts joint orientations to be closer to the desired pose within this constraint. For a new rotation quaternion r_t, previous rotation r_{t-1}, desired rotation d_t, and the derivative in the source animation of the desired frame $\triangle d_t = d_t \bar{d}_{t-1}$, the orientation of a joint at frame t is given by

$$r_t = \text{slerp}\left(r_{t-1}, d_t; \frac{\text{angle}(\triangle d_t)}{\text{angle}(d_t \bar{r}_{t-1})}\right),$$

where "slerp" is the quaternion spherical interpolation function [Shoemake 1985], and "angle" gives the angle of the axis-angle representation of a quaternion. We found that using world-space rather than parent-space orientation in the desired and previous pose produces more realistic results and preserves the "feel" of the desired animation.

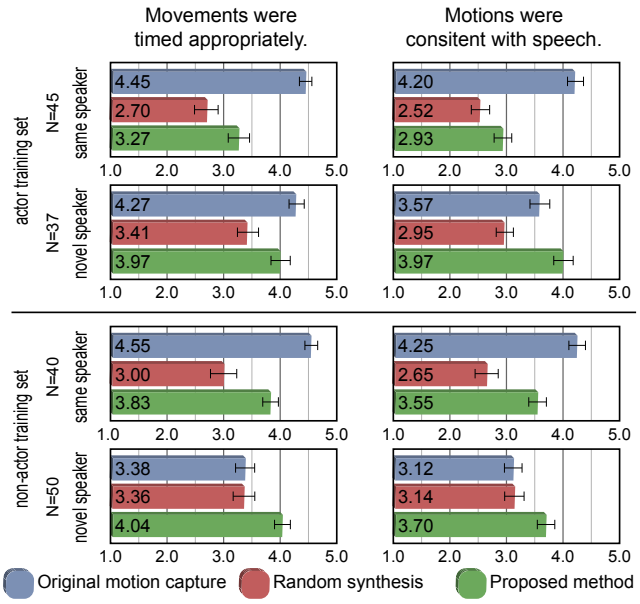


Figure 8: *Mean scores on a 5-point Likert scale for the four evaluation surveys. Error bars indicate standard error.* N *indicates the number of responders.*

10 Results

There is no single correct body language sequence for a given utterance, which makes validation of our system inherently difficult. The only known way to determine the quality of synthesized body language is by human observation. To this end, we conducted four surveys to evaluate our method. Participants were asked to evaluate three animations corresponding to the same utterance, presented in random order on a web page. The animations were mapped onto a simple generic character, as show in Figure 6. The responders were recruited from a broad group university students unfamiliar with the details of the system. Student ID numbers were used to screen out repeat responders. The utterances ranged from 40 to 60 seconds in length. In addition to the synthesized sequence, two controls were used. One of the controls contained motion capture of the original utterance being spoken. The other control was generated by randomly selecting new animation segments whenever the current segment terminated, producing an animation that did not correspond to the speech but still appeared generally coherent. The original motion capture provides a natural standard for quality, while random selection represents a simple alternative method for synthesizing body language in real time. Random selection has previously been used to animate facial expressions [Perlin 1997] and to add detail to coarsely defined motion [Li et al. 2002].

Two sets of training data were used in the evaluation, from two different speakers. The first training set consisted of 30 minutes of motion capture data in six scenes, recorded from a trained actor. The second training set consisted of 15 minutes in eight scenes, recorded from a good speaker with no special training. For each training set, two surveys were conducted. The "same speaker" surveys sought to determine the quality of the synthesized animation compared to motion capture of the original speaker, in order to ensure a reliable comparison without interference from confounding variables, such as gesturing style. These surveys used utterances from the training speakers that were not present in the training data. The "novel speaker" surveys sought to determine how well the system generalized to different speakers.

Participants rated each animation on a five-point Likert scale for timing and appropriateness. The questions and average scores for the surveys are presented in Figure 8. In all surveys, the synthesized sequence outperformed the random sequence, according to a pairwise, single-tailed t-test with $p < 0.05$. In fact, with the exception of the actor trained same speaker survey, the score given to the synthesized sequences remained quite stable. The relatively low scores of both the synthesized and randomly constructed sequences in this survey may be accounted for by the greater skill of the trained actor. In both of the novel speaker surveys, the original motion capture does not always outperform the synthesized sequence, and in one of the surveys, its performance is comparable to that of the random sequence. This indicates that the two speakers used in the generalization tests had body language that was not as compelling as the training data. This is not unexpected, since skilled speakers were intentionally chosen to create the best possible training data. The stability of synthesized sequence scores, even for the novel speakers, indicates that our system was able to successfully transplant the training speakers' more effective gesturing styles onto the novel speakers' voices.

All the videos used in the surveys are included with the supplementary material, available through the ACM Digital Library. All examples shown in this paper and the accompanying video were generated using the actor training set.

11 Discussion and Future Work

We presented a system for generating expressive body language animations for human characters in real time from live speech input. The system is efficient enough to run on a 2 GHz Intel Centrino processor, making it suitable for modern consumer PCs. Our method works off of two assumptions: the co-occurrence of gestures and syllable peaks and the usefulness of prosody for identifying body language. The former assumption is justified by the "synchrony rule" [McNeill 1992], and the latter by the relationship between prosody and emotion [Adolphs 2002], and by extension body language [Wallbot 1998; Montepare et al. 1999]. The effectiveness of our method was validated by a comparative study that confirmed that these assumptions produce plausible body language that compares well to the actual body language accompanying the utterance.

In addition to the survey discussed in Section 10, we conducted a pilot study in which participants were additionally shown an an-

imation with randomly selected gestures synchronized to syllable peaks, and an animation generated with our system but without the synchrony rule (i.e., all gestures ran to completion). Although these animations were not included in the final survey to avoid fatiguing the responders, comments from the pilot study indicated that both synchrony and proper selection were needed to synthesize plausible body language. Responders noted that the character animated with random selection "didn't move with the rhythm of the speech," while the character that did not synchronize with syllable peaks appeared "off sync" and "consistently low energy." These findings suggest that both the "synchrony rule" and prosody-based gesture selection are useful for gesture synthesis. The accompanying video also contains examples of gestures synthesized without the synchrony rule and randomly selected gestures that are synchronized to the speech.

Despite the effectiveness of this method for synthesizing plausible animations, it has several inherent limitations. Most importantly, relying on prosody alone precludes the system from generating meaningful iconic gestures when they are not accompanied by emotional cues. Metaphoric gestures are easier to select because they originate from a smaller repertoire and are more abstract [McNeill 1992], and therefore more tolerant of mistakes, but iconic gestures cannot be "guessed" without interpreting the words in an utterance. This fundamental limitation may be addressed in the future with a method that combines rudimentary word recognition with prosody-based gesture selection. Word recognition may be performed either directly or from phonemes, which can already be extracted in real time [Park and Ko 2008]. Additional work would be necessary, however, to extract meaning in a *predictive* manner, since gestures often precede or coincide with the co-occuring word, rather than following it [McNeill 1992].

A second limitation of our method is that it must synthesize gestures from information that is already available to the listener, thus limiting its ability to provide supplementary details. While there is no consensus in the linguistics community on just how much information is conveyed by gestures [Krauss et al. 1995; Loehr 2004], a predictive speech-based real-time method is unlikely to impart on the listener any more information than could be obtained from simply listening to the speaker attentively, while real gestures often convey information not present in speech [McNeill 1992]. Therefore, while there are clear benefits to immersiveness and realism from compelling and automatic animation of human-controlled characters, such methods cannot provide additional details without some form of additional input. This additional input, however, need not necessarily be as obtrusive as keyboard or mouse controls. For example, facial expressions carry more information about emotional state than prosody [Adolphs 2002], which suggests that more informative gestures may be synthesized by analyzing the speaker's facial expressions, for example though a consumer webcam.

A third limitation of the proposed method is its inherent tendency to overfit the training data. While prosody is useful in selecting appropriate gestures, it is unlikely to provide enough information to decide on some aspects of body language, such as the form of semantically tied gestures. When such gestures are present in the training data, they may be incorrectly associated with prosody features. A sufficiently large amount of training data could remedy this problem. However, a more sophisticated approach that only models those aspects of gesture that are expected to correlate well with prosody could perform better.

Besides addressing the limitations of the proposed system, future work may also expand its capabilities. Training data from multiple individuals, for example, may allow the synthesis of "personalized" gesture styles, as advocated by [Neff et al. 2008]. A more advanced method of extracting gestures from training data may allow for the principal joints of a gesture to be identified automatically, which would both eliminate the need for the current separation of leg, arm, and head gestures and allow for more intelligent blending of gestures with other animations. This would allow a gesturing character to engage in other activities with realistic interruptions for performing important gestures.

In addition to animating characters, we hope our method will eventually reveal new insight into how people form and perceive gestures. Our pilot survey already suggests that our method may be useful for confirming the validity of the synchrony rule and the relationship between prosody and gesture. Further work could explore the relative importance of various gesture types, as well as the impact of timing and appropriateness on perceived gesture quality.

As presented in the accompanying video, our method generates plausible body language for a variety of utterances and speakers. Prosody allows our method to detect emphasis and produce gestures that reflect the emotional state of the speaker. The system generates compelling body language from a live speech stream and does not require specialized input, making it particularly appropriate for animating human characters in networked virtual worlds.

12 Acknowledgments

We thank Alison Sheets, Stefano Corazza and the staff of Animotion Inc., Jeffrey Lee and the staff of PhaseSpace Inc., and Chris Bregler for assisting us with motion capture acquisition and processing, Sebastián Calderón Bentin for being our motion capture actor, Chris Platz for help with illustrations, and Jerry Talton for additional work. This work was supported by NSF grants SES-0835601 and CCF-0641402.

References

ADOLPHS, R. 2002. Neural systems for recognizing emotion. *Current Opinion in Neurobiology 12*, 2, 169–177.

ALBRECHT, I., HABER, J., AND PETER SEIDEL, H. 2002. Automatic generation of non-verbal facial expressions from speech. In *Computer Graphics International*, 283–293.

ARIKAN, O., AND FORSYTH, D. A. 2002. Interactive motion generation from examples. *ACM Transactions on Graphics 21*, 3, 483–490.

BARBIČ, J., SAFONOVA, A., PAN, J.-Y., FALOUTSOS, C., HODGINS, J. K., AND POLLARD, N. S. 2004. Segmenting motion capture data into distinct behaviors. In *Proceedings of Graphics Interface 2004*, 185–194.

BESKOW, J. 2003. *Talking Heads - Models and Applications for Multimodal Speech Synthesis*. PhD thesis, KTH Stockholm.

BOERSMA, P. 1993. Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound. In *Proceedings of the Institute of Phonetic Sciences*, vol. 17, 97–110.

BOERSMA, P. 2001. Praat, a system for doing phonetics by computer. *Glot International 5*, 9-10, 314–345.

BRAND, M. 1999. Voice puppetry. In *Proc. ACM SIGGRAPH*, 21–28.

BREGLER, C., COVELL, M., AND SLANEY, M. 1997. Video rewrite: driving visual speech with audio. In *SIGGRAPH '97: ACM SIGGRAPH 1997 Papers*, ACM, New York, NY, USA, 353–360.

CASSELL, J., PELACHAUD, C., BADLER, N., STEEDMAN, M., ACHORN, B., BECKET, T., DOUVILLE, B., PREVOST, S., AND STONE, M. 1994. Animated conversation: rule-based generation of facial expression, gesture & spoken intonation for multiple conversational agents. In *SIGGRAPH '94: ACM SIGGRAPH 1994 Papers*, ACM, New York, NY, USA, 413–420.

CASSELL, J., VILHJÁLMSSON, H. H., AND BICKMORE, T. 2001. Beat: the behavior expression animation toolkit. In *Proc. ACM SIGGRAPH*, 477–486.

CASSELL, J. 2000. Embodied conversational interface agents. *Communications of the ACM 43*, 4, 70–78.

CHUANG, E., AND BREGLER, C. 2005. Mood swings: expressive speech animation. *ACM Transactions on Graphics 24*, 2, 331–347.

EZZAT, T., GEIGER, G., AND POGGIO, T. 2002. Trainable videorealistic speech animation. In *SIGGRAPH '02: ACM SIGGRAPH 2002 Papers*,

ACM, New York, NY, USA, 388–398.

FOD, A., MATARIC, M. J., AND JENKINS, O. C. 2002. Automated derivation of primitives for movement classification. *Autonomous Robots 12*, 39–54.

HARTMANN, B., MANCINI, M., AND PELACHAUD, C. 2002. Formational parameters and adaptive prototype instantiation for mpeg-4 compliant gesture synthesis. In *Proceedings on Computer Animation*, IEEE Computer Society, Washington, DC, USA, 111.

JENSEN, C., FARNHAM, S. D., DRUCKER, S. M., AND KOLLOCK, P. 2000. The effect of communication modality on cooperation in online environments. In *Proceedings of CHI '00*, ACM, New York, NY, USA, 470–477.

JU, E., AND LEE, J. 2008. Expressive facial gestures from motion capture data. *Computer Graphics Forum 27*, 2, 381–388.

KENDON, A. 2004. *Gesture – Visible Action as Utterance*. Cambridge University Press, New York, NY, USA.

KIPP, M., NEFF, M., AND ALBRECHT, I. 2007. An annotation scheme for conversational gestures: how to economically capture timing and form. *Language Resources and Evaluation 41*, 3-4 (December), 325–339.

KOPP, S., AND WACHSMUTH, I. 2004. Synthesizing multimodal utterances for conversational agents: Research articles. *Computer Animation and Virtual Worlds 15*, 1, 39–52.

KOVAR, L., GLEICHER, M., AND PIGHIN, F. 2002. Motion graphs. *ACM Transactions on Graphics 21*, 3, 473–482.

KRAUSS, R. M., DUSHAY, R. A., CHEN, Y., AND RAUSCHER, F. 1995. The communicative value of conversational hand gestur. *Journal of Experimental Social Psychology 31*, 6, 533–552.

LAFFERTY, J. D., MCCALLUM, A., AND PEREIRA, F. C. N. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of the 18th International Conference on Machine Learning*, Morgan Kaufmann Inc., 282–289.

LEE, J., CHAI, J., REITSMA, P. S. A., HODGINS, J. K., AND POLLARD, N. S. 2002. Interactive control of avatars animated with human motion data. *ACM Transactions on Graphics 21*, 3, 491–500.

LI, Y., AND SHUM, H.-Y. 2006. Learning dynamic audio-visual mapping with input-output hidden markov models. *IEEE Transactions on Multimedia 8*, 3, 542–549.

LI, Y., WANG, T., AND SHUM, H.-Y. 2002. Motion texture: a two-level statistical model for character motion synthesis. In *SIGGRAPH '02: ACM SIGGRAPH 2002 Papers*, ACM, New York, NY, USA, 465–472.

LOEHR, D. 2004. *Gesture and Intonation*. PhD thesis, Georgetown University.

MAERAN, O., PIURI, V., AND STORTI GAJANI, G. 1997. Speech recognition through phoneme segmentation and neural classification. *Instrumentation and Measurement Technology Conference, 1997. IMTC/97. Proceedings. 'Sensing, Processing, Networking.', IEEE 2* (May), 1215–1220.

MAJKOWSKA, A., ZORDAN, V. B., AND FALOUTSOS, P. 2006. Automatic splicing for hand and body animations. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 309–316.

MCNEILL, D. 1992. *Hand and Mind: What Gestures Reveal About Thought*. University Of Chicago Press.

MCNEILL, D. 2005. *Gesture and Thought*. University Of Chicago Press, November.

MONTEPARE, J., KOFF, E., ZAITCHIK, D., AND ALBERT, M. 1999. The use of body movements and gestures as cues to emotions in younger and older adults. *Journal of Nonverbal Behavior 23*, 2, 133–152.

MORENCY, L.-P., SIDNER, C., LEE, C., AND DARRELL, T. 2007. Head gestures for perceptual interfaces: The role of context in improving recognition. *Artificial Intelligence 171*, 8-9, 568–585.

MÜLLER, M., RÖDER, T., AND CLAUSEN, M. 2005. Efficient content-based retrieval of motion capture data. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, ACM, New York, NY, USA, vol. 24, 677–685.

NEFF, M., KIPP, M., ALBRECHT, I., AND SEIDEL, H.-P. 2008. Gesture modeling and animation based on a probabilistic re-creation of speaker style. *ACM Transactions on Graphics 27*, 1, 1–24.

PARK, J., AND KO, H. 2008. Real-time continuous phoneme recognition system using class-dependent tied-mixture hmm with hbt structure for speech-driven lip-sync. *IEEE Transactions on Multimedia 10*, 7 (Nov.), 1299–1306.

PERLIN, K., AND GOLDBERG, A. 1996. Improv: a system for scripting interactive actors in virtual worlds. In *SIGGRAPH '96: ACM SIGGRAPH 1996 Papers*, ACM, 205–216.

PERLIN, K. 1997. Layered compositing of facial expression. In *SIGGRAPH '97: ACM SIGGRAPH 97 Visual Proceedings*, ACM, New York, NY, USA, 226–227.

RABINER, L. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE 77*, 2, 257–286.

RAMSHAW, L. A., AND MARCUS, M. P. 1995. Text chunking using transformation-based learning. In *Proceedings of the Third Workshop on Very Large Corpora*, 82–94.

SARGIN, M. E., ERZIN, E., YEMEZ, Y., TEKALP, A. M., ERDEM, A., ERDEM, C., AND OZKAN, M. 2007. Prosody-driven head-gesture animation. In *ICASSP '07: IEEE International Conference on Acoustics, Speech, and Signal Processing*.

SCHERER, K. R., BANSE, R., WALLBOTT, H. G., AND GOLDBECK, T. 1991. Vocal cues in emotion encoding and decoding. *Motivation and Emotion 15*, 2, 123–148.

SCHRÖDER, M. 2004. *Speech and Emotion Research: An overview of research frameworks and a dimensional approach to emotional speech synthesis*. PhD thesis, Phonus 7, Research Report of the Institute of Phonetics, Saarland University.

SHOEMAKE, K. 1985. Animating rotation with quaternion curves. In *SIGGRAPH '85: ACM SIGGRAPH 1985 Papers*, ACM, New York, NY, USA, 245–254.

SIFAKIS, E., SELLE, A., ROBINSON-MOSHER, A., AND FEDKIW, R. 2006. Simulating speech with a physics-based facial muscle model. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 261–270.

STONE, M., DECARLO, D., OH, I., RODRIGUEZ, C., STERE, A., LEES, A., AND BREGLER, C. 2004. Speaking with hands: creating animated conversational characters from recordings of human performance. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, ACM, New York, NY, USA, 506–513.

TERKEN, J. 1991. Fundamental frequency and perceived prominence of accented syllables. *The Journal of the Acoustical Society of America 89*, 4, 1768–1777.

TREUILLE, A., LEE, Y., AND POPOVIĆ, Z. 2007. Near-optimal character animation with continuous control. In *SIGGRAPH '07: ACM SIGGRAPH 2007 Papers*, ACM, New York, NY, USA.

WALLBOT, H. G. 1998. Bodily expression of emotion. *European Journal of Social Psychology 28*, 6, 879–896.

WANG, J., AND BODENHEIMER, B. 2008. Synthesis and evaluation of linear motion transitions. *ACM Transactions on Graphics 27*, 1 (Mar), 1:1–1:15.

WARD, J. H. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association 58*, 301, 236–244.

XUE, J., BORGSTROM, J., JIANG, J., BERNSTEIN, L., AND ALWAN, A. 2006. Acoustically-driven talking face synthesis using dynamic bayesian networks. *IEEE International Conference on Multimedia and Expo* (July), 1165–1168.