EUROGRAPHICS 2008 / G. Drettakis and R. Scopigno
(Guest Editors)

Volume 27 (2008), Number 2

Expressive Facial Gestures From Motion Capture Data

Eunjung Ju and Jehee Lee

Seoul National University†

Abstract

Human facial gestures often exhibit such natural stochastic variations as how often the eyes blink, how often the eyebrows and the nose twitch, and how the head moves while speaking. The stochastic movements of facial features are key ingredients for generating convincing facial expressions. Although such small variations have been simulated using noise functions in many graphics applications, modulating noise functions to match natural variations induced from the affective states and the personality of characters is difficult and not intuitive. We present a technique for generating subtle expressive facial gestures (facial expressions and head motion) semi-automatically from motion capture data. Our approach is based on Markov random fields that are simulated in two levels. In the lower level, the coordinated movements of facial features are captured, parameterized, and transferred to synthetic faces using basis shapes. The upper level represents independent stochastic behavior of facial features. The experimental results show that our system generates expressive facial gestures synchronized with input speech.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Three-Dimensional Graphics and Realism]: Animation

1. Introduction

Human facial expression is dynamic in nature, often exhibiting natural stochastic variations. The emotional state and individual personality are closely related to such natural variations as how often the eyes blink, how often the eyebrows and the nose twitch, and how the head moves while speaking. The stochastic movements of facial features are key ingredients for creating expressive conversational agents.

Virtual characters in many video games and virtual environments often employ noise functions [Per95, Per07] to incorporate noise-like subtle movements into facial expression. Adding smooth temporal noise patterns makes an animated character look less like a mannequin. However, modulating noise functions to simulate natural variations induced from the emotional state and the personality of characters is neither easy nor intuitive.

An alternative approach we are pursuing is to capture the stochastic variational patterns from live human actors and use the recorded patterns to add life-like subtle movements into a synthetic face and its head motion. Such stochastic patterns are typically modelled as Markov random fields, which pose several challenges for addressing issues specific to facial expression. Facial features sometimes move in a strongly coordinated manner in order to convey specific expressions, whereas some facial features look rather

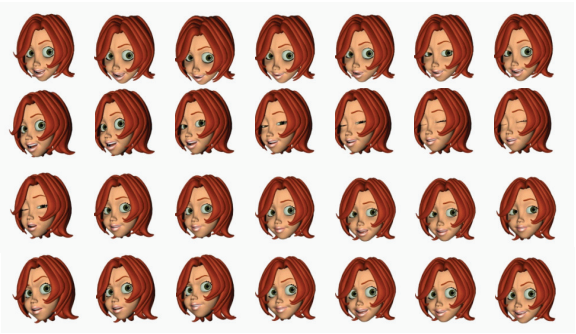

Figure 1: *Expressive facial expressions and head motions are captured, parameterized, and transferred to the synthetic face.*

† e-mail: {ejjoo,jehee}@mrl.snu.ac.kr

independent and stochastic. For example, a smiling face is a combination of smiling eyes, brows, and a mouth that pose and move in harmony. On the other hand, small expressive variations of eye brows, eye blinking, and the head motion are stochastic and not strongly coordinated with the other facial features. These coordinated facial gestures and independent facial gestures should be considered simultaneously. We found that the subtle change of facial expressions in conversation is often synchronous with the change of acoustic features such as the intensity of speech sounds rather than more detailed prosodic features such as phonemes.

In this work, we present an intuitive, easy-to-use facial animation system that generates expressive facial gestures semi-automatically. Even though face motion capture data are used for synthesizing the stylistic variations and mood of facial expression, the animated face is not necessarily human-like, but can be extremely exaggerated or even cartoon-like. Recorded speech data can also be employed to generate synchronized lip movements and drive the stochastic process of facial feature movements. Given any speech data, our system allows the user to create lively animated talking faces in a few minutes.

2. Related Work

Since Perlin [Per85] introduced his well-known noise function in 1985, the noise function has been employed in a wide variety of graphics applications. In particular, Perlin and his colleagues used the noise function to make animated characters and their faces appear more natural by adding small variations in their movements [Per95, PG96]. Similar ideas have been exercised in a wide variety of applications such as personalized idle motion synthesis [EMMT04].

A great deal of previous research on facial animation involve lip-synching, which refers to synthesizing facial motion that is synchronized with input speech [Bra99, BCS97, CTFP05, CB05, **?**, DN06, EGP02, KMT03]. Most approaches used phonemes as elementary speech units and intended to generate realistic lip movements driven by a sequence of phonemes. Our synthetic faces are also driven by input speech, but the generation of realistic lip movements is not a major focus of this work. The goal of this work is to reproduce small variations in facial gestures (including facial expressions and head motion) that convey the affective states, mood, and personality of the character. The strong interrelation between facial gestures and prosodic features has been reported in the speech processing literature [BN07, BDG*07, YRVB98]. However, the interrelation between facial gestures and individual phonemes is not obvious. Our main focus is to synthesize facial gestures possibly driven by acoustic features of input speech. For the fidelity of the resulting facial animation, supplementary lip movements are generated separately and blended in later by analyzing phonemes from input speech.

Figure 2: *Facial motion capture.*

Facial expression synthesis has been extensively explored in the computer graphics community. Facial expressions can be procedurally defined [BB02, CPB*94, ZLGS03], physically simulated [LTW95, SSRMF06], or defined using a basis of shapes [CB05, JTDP03, PKC*03]. Our approach makes use of basis shapes for capturing and synthesizing facial expressions because of their versatility and flexibility. The basis shapes can be manually defined [PKC*03, ZLGS03] or automatically identified from a stream of training data via component analysis [CTFP05, CXH03, MKPG05] or factorization [CB05, VBPP05]. Although existing methods can generate convincing static facial expressions and simulate the short-term dynamics of facial action units successfully, only a few of them addressed the problem of capturing and simulating the temporal stochastic movements of facial features in extended expressive motions.

A popular way of modeling temporal stochastic behavior is based on Markov processes, which were employed for animating fullbody motions using motion databases [LCR*02]. The Markov process with motion data entails data structures, called *motion graphs*, which can also be used for synthesizing facial animation [CTFP05, ZSCS04]. We present a novel two-level representation based on Markov random fields. In the lower level, the coordinated movements of facial features are captured, parameterized, and transferred to synthetic faces using basis shapes. In the upper level, independent stochastic behavior of facial features are simulated.

3. Data Collection and Processing

To acquire realistic human facial motion data, we used a Vicon optical motion capture system. 12 cameras tracked 68 retro-reflection markers on the face and 7 markers on the head at the rate of 120 frames/second (see Figure 2). Facial motion data thus captured are then down-sampled to 30 frames/second for realtime display. We recorded 12 subjects reading fairy tales and conversing freely at 6 different emotional states (neutral, depressed, delighted, annoyed, exaggerated, exasperated) for about a minute for each. We also captured static facial expressions (neutral, sad, happy, afraid, angry, surprise) for each subject. The rigid transform of the head motion is computed using 7 head markers and the coordinates of 68 facial markers are represented with respective to the head reference system. Three of our subjects are

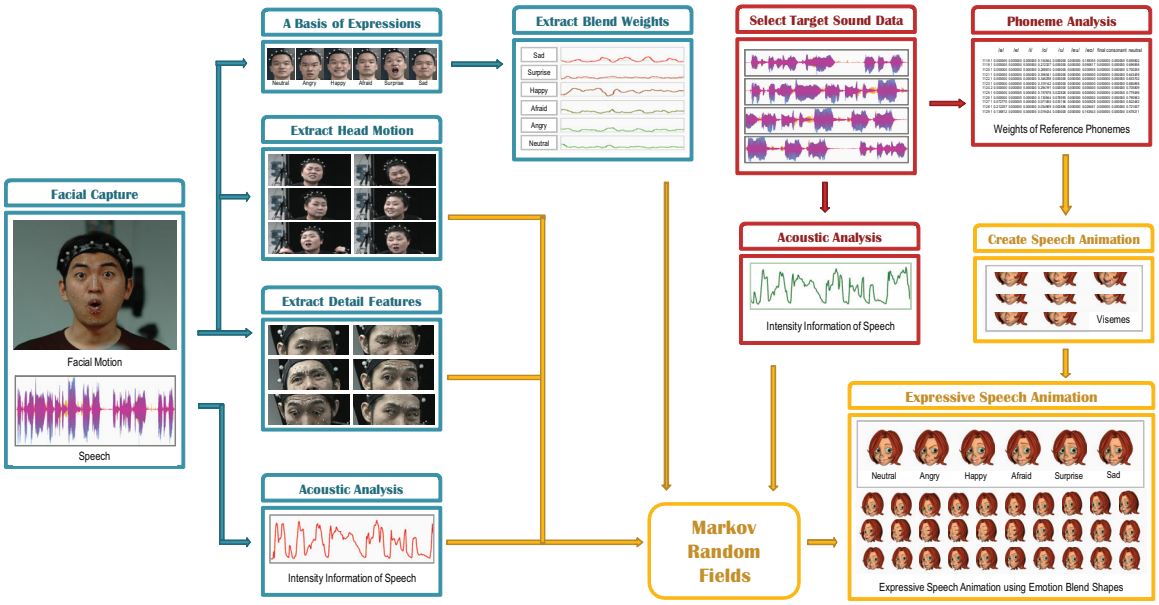

Figure 3: *The overview of our facial gesture synthesis system.*

professional (two actors and a narrator) and all the others are non-professional volunteers. The speech sound data were recorded simultaneously in the motion capture session at the rate of 32KHz. We used Praat software [BW07] to analyze the pitch and loudness of speech data.

4. Facial Gestures From Motion Capture

We would like to identify expressive facial gestures independently of the content (utterance of sentences and the corresponding lip movements) of facial motion capture data. The facial gestures thus obtained are used to animate synthetic faces uttering different contents (see Figure 3). Simply ignoring markers on the lips is not an ideal solution, because the shape and movement of lips retain a lot of information for capturing and transferring expressive gestures.

The facial expression $F(t)$ captured at frame t is represented as a long vector that concatenates the coordinates of marker points.

$$F(t) = R(t)\left(N + \sum_i c_i(t)B_i + P(t) + \sum_j D_j(t)\right) + T(t),$$

where $R(t)$ and $T(t)$ represent the rigid transform of the head motion, N is a neutral expression, B_i's are a basis of facial expressions, $P(t)$ is lip movements, and $D_j(t)$ is the detailed movements of facial features. Each element of B_i is actually a displacement from the neutral expression, so $N + B_i$ for each i represents an individual facial expression in a certain emotional state and c_i are the weight values of basis expressions. In our experiments, six basis expressions are used (see Figure 4).

Among the terms in the above equation, $R(t)$, $T(t)$, N and B_i are determined trivially in the motion capture session. We would like to determine two terms, $c_i(t)$ and $D_j(t)$, independently of $P(t)$. Blend weight $c_i(t)$ for each i affects the entire facial region and thus encodes a coordinated facial expression that requires a harmony of synchronized facial features. On the other hand, independent feature $D_j(t)$ for each j acts on a small portion of the facial region and thus encodes the stochastic movements of an individual facial feature. $P(t)$ is considered as a sort of independent features that generates lip-synching. We identified five non-overlapping regions (one for $P(t)$ and the others for $D_j(t)$) and sorted corresponding facial markers into groups (see Figure 5).

4.1. Determining blend weights

Given the configuration of marker points F for frame t, weight values c_i's are determined such that $\{R^{-1}(F-T)-N\}-\sum_i c_i B_i$ is minimized. The weight values thus obtained are quite noisy because of small variations induced from subtle facial gestures and uttering sentences. In order to cancel out small variations and lip movements, we consider the average configuration $\bar{F} = \big(F(t-k)+\cdots+F(t+k)\big)/(2k+1)$ of neighboring frames. In our experiments, the window size is $2k+1 = 41$ (1.36 seconds). We use $\bar{F}$ instead of F to determine the weight values. The residual $F-\bar{F}$ is considered as stochastic movements that will be encoded in P and D_j.

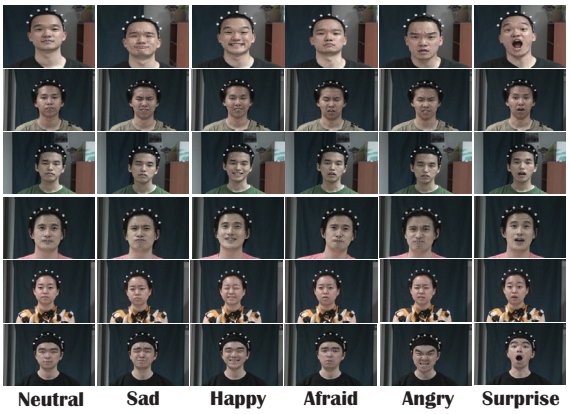

Figure 4: *The basis of facial expressions in motion capture data.*

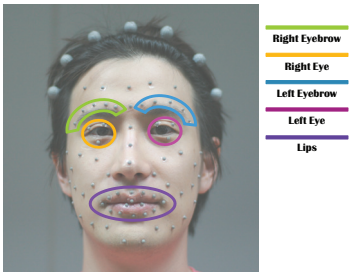

Figure 5: *Groupings of facial markers. Each group corresponds to either P or D_j.*

This parameter estimation problem allows a least squares solution, which often provides us with extremely extrapolated results. Such extrapolated blend weights are undesirable for transferring facial expressions from motion capture data to synthetic faces. In order to avoid extreme extrapolation, the range of each weight should be constrained such that $-\varepsilon \le c_i \le 1+\varepsilon$. If $\varepsilon = 0$ and $\sum_i c_i \le 1$, facial expressions are represented as a convex combination of basis expressions. In practice, the convexity condition is so restrictive that some exaggerated expressions cannot be reproduced faithfully. If ε is too large, the weight values tend towards noisy. In our experiments, we set $0 \le \varepsilon \le 1$.

With this constraint, the least squares problem is not straightforward any more. We determine the weight values approximately by sampling the parameter space. We generate random weight values $\{c_i^k | i = 1,\cdots,n_b, k = 1,\cdots,n_s\}$ satisfying the constraint and produce a set of facial expressions $\{F_k | k = 1,\cdots,n_s\}$, where n_s is the number of random samples and n_b is the number of basis expressions.

$$F_k = \sum_{i=1}^{n_b} c_i^k B_i.$$

Given any novel facial expression F, its n_k-nearest neighbors from the set of random samples are used to estimate its weight values. Without loss of generality, we assume that the first n_k samples are nearest to F. Then, F can be estimated as a combination of its neighboring samples.

$$R^{-1}(\bar{F}-T)-N \simeq \sum_{k=1}^{n_k} a_k F_k$$
$$= \sum_k a_k (\sum_i c_i^k B_i) = \sum_i (\sum_k a_k c_i^k) B_i = \sum_i c_i B_i$$

where a_k is inversely proportional to the distance between $R^{-1}(\bar{F}-T)-N$ and F_k and normalized such that $\sum_k a_k = 1$. Hence, $\bar{F}$ can be represented approximately as a combination of basis expressions with weights $c_i = \sum_k a_k c_i^k$. In order to locate nearest neighbors efficiently, we store data in a *k*d-tree and search n_k-nearest neighbors approximately within a small error bound using ANN library [MA06].

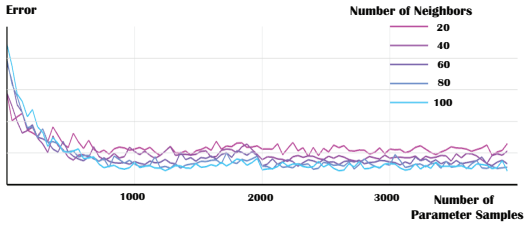

Figure 6: *The plot of approximation errors with respect to the number of samples and the size of nearest neighborhood selection.*

Our experimental results in Figure 6 show that the approximation error decreases gracefully with respect to the number of random samples. The number of nearest neighbors participating in the estimation process also affects the convergence of approximation errors. In our experiments, we generated 1000 random samples and selected 60 nearest neighbors to estimate weight values.

4.2. Determining independent stochastic features

Once the blend weights of basis expressions are determined, independent features D_j's and lip movements P can be determined simply as the residual of blend shape approximation. D_j and P are long vectors that concatenates the coordinates of marker points. Most of their elements are zero and non-zero values are only for the markers in the corresponding marker group (see Figure 5). Since the marker groups of P and D_i's are non-overlapping,

$$R^{-1}\left(F-T\right)-N-\sum_i c_i B_i = \left(P+\sum_j D_j+\varepsilon\right)$$

determines their values. The error $\varepsilon(t)$ is the movement of markers that are not included in any of marker groups. $\varepsilon(t)$

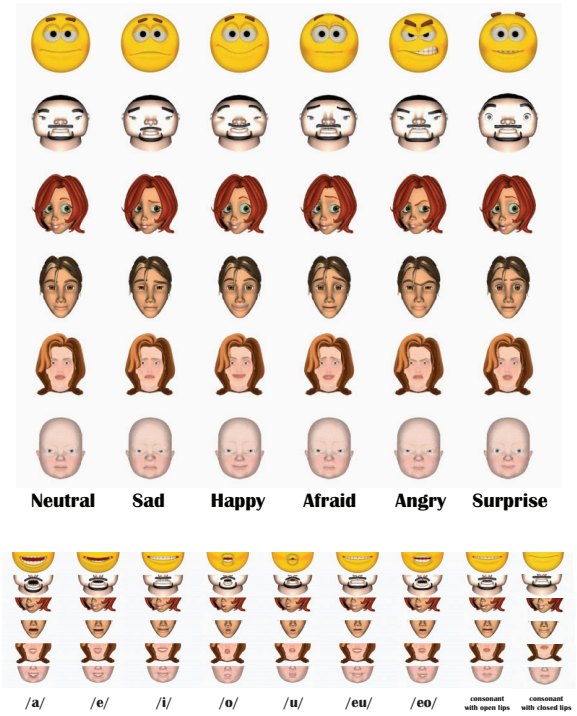

Figure 7: *(Top) The basis of synthetic facial expressions and (Bottom) The basis of visemes.*

is, in practice, not clearly noticeable in facial animation and thus we simply ignore the error term.

5. Facial Gesture Synthesis

In this section, we discuss how to transfer the natural variations of facial gestures obtained from human motion to synthetic faces. The synthetic face can be driven by input speech. The driving speech signal is, in general, different from the content we recorded in the motion capture session. We intend to synthesize facial gestures independently of language, syllables, and phonemes except for lip movements. Lip synching is handled separately.

The synthetic facial expression $\tilde{F}(t)$ of our character at frame t is represented as a vector that concatenates the coordinates of mesh points in the face geometry. We use a tilde to denote symbols related to synthetic faces.

$$\tilde{F}(t) = \tilde{R}(t)\left(\tilde{N} + \sum_i \tilde{c}_i(t)\tilde{B}_i + \sum_l \tilde{d}_l(t)\tilde{P}_l + \sum \tilde{D}_j(t)\right) + \tilde{T}(t)$$

where $\tilde{R}(t)$ and $\tilde{T}(t)$ represent the rigid transform of the head motion. $\tilde{N}$ is a static, neutral expression of our face mesh. $\tilde{B}_i$'s are a basis of facial expressions and $\tilde{c}_i$'s are their weight values. $\tilde{P}_l$'s are a basis of visemes and $\tilde{d}_l$'s are their weight values.

Among the terms in the above equation, $\tilde{N}$, $\tilde{B}_i$ are $\tilde{P}_l$ are static geometric models manually designed by artists (see Figure 7). d_l's are determined from the phoneme analysis of the driving speech signal. All the other terms are determined in the process of Markov random fields.

5.1. Markov Random Fields

In this section, we discuss how to determine a sequence of blend weight $\tilde{c}_i(t)$ using Markov random fields. Other terms $\tilde{R}(t)$, $\tilde{T}(t)$, $\tilde{D}_j(t)$ can be computed similarly.

Assume that the first $t-1$ frames $\tilde{c}_i(1),\cdots,\tilde{c}_i(t-1)$ has already been computed and we want to determine the next frame $\tilde{c}_i(t)$. The Markov random fields method determines $\tilde{c}_i(t)$ by matching its h-preceding frames $\{\tilde{c}_i(t-1),\cdots,\tilde{c}_i(t-h)\}$ to reference data. The probability of selecting $c_i(\hat{t})$ as the value of $\tilde{c}_i(t)$ is proportional to $\exp(-\mathrm{dist}(t,\hat{t})/\sigma)$, where

$$\mathrm{dist}(t,\hat{t}) = \sum_{j=1}^{h}\left(\alpha|\tilde{S}(t-j) - S(\hat{t}-j)|^2 + \sum_{i=1}^{n_b}|\tilde{c}_i(t-j) - c_i(\hat{t}-j)|^2\right).$$

σ controls the mapping between the distance measure and the probability of selection. $S(\cdot)$ is the intensity of reference speech data recorded synchronized with facial motion capture data and $\tilde{S}(\cdot)$ is the intensity of speech data that the synthetic face utters. The first term favors the match of speech and the second term favors the match of facial expressions at previous frames. α weighs the importance of two terms. If α is high, facial gestures synchronize well with the speech data. The low value of α tends to favor the smooth transitioning of facial gestures over lip synching.

At every frame of the synthesis process, the probability of transitioning from the current frame to every other frames in the reference data can be computed to select the next frame probabilistically. This approach allows bad transitions to happen, though their probabilities are pretty low. In practice, such bad transitions are sometimes disturbing. Instead, we search a small number (2 to 4 in our experiments) of good candidates $\hat{t}$ for transitioning that minimize $\mathrm{dist}(t,\hat{t})$ and select one of them based on their probabilities.

The window size h is selected empirically. A small value for h permits the flexibility in transitioning, while a larger value weighs better context matching over the variety of output animation. In our experiments, $h = 10$ was a nice trade-off between the variety and quality of the resulting animation.

5.2. Transferring Detailed Features

The detailed facial features are dependent on the face geometry. Some of our face models are exaggerated or even

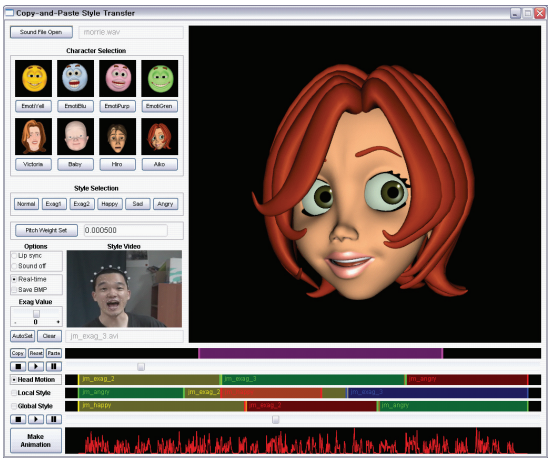

Figure 8: *The user interface of our facial animation system*

cartoon-like. Therefore, transferring D_j to $\tilde{D}_j$ requires extra efforts for establishing correspondences between facial markers and synthetic face geometry. We manually picked locations on the face geometry corresponding to facial markers. The movements of facial markers in D_j are then scaled appropriately to match the geometry of synthetic faces. For example, we found an affine transform between actual marker locations on the eye lids and their corresponding points on the synthetic face such that they match at wide opening and closing of eye lids.

Once the correspondences are established, RBF (radial basis function) interpolation [TO99] generates deformed face geometry $\tilde{D}_j$ corresponding to D_j. In order to localize the influence region of the deformation, bell-shaped cubic B-spline basis functions are employed. The linear polynomial approximation term is not used because its support region is infinitely wide.

6. Experimental Results

We recorded a variety of facial gestures from 12 subjects. In the motion capture session, each subject was instructed to read fairy tales and talk freely in several different emotional states. We also recorded each subject exhibiting his/her own distinctive personality in their facial expressions as much as possible. Each motion clip is about one minute long and contains an individual "style" of facial gestures. A set of such motion clips is used as a palette of facial gesture styles.

User interface. Our system provides the user with an easy-to-use user interface for creating expressive facial animations in a few minutes (see Figure 8). In the workflow, the user first imports input speech data and a palette of facial gesture styles. Then, the user can "paint" the gesture styles on the timeline synchronized with the input speech. It is also possible to use different styles for different features. For example, the eyebrows can imitate the gestures of subject A, while the head motion is simulating the style of subject B.

Facial expression transfer. Captured facial expressions can be directly transferred to a synthetic face, even though the synthetic face does not resemble the motion capture subject at all (see Figure 9). It is possible because basis expressions are used on both sides [PKC*03]. Captured expressions and transferred animated expressions share the same blend weights, while their bases are totally different. With this idea, we can manipulate synthetic faces for computer puppetry via real time motion capture.

Comparison to noise functions. Our animated result easily outperforms the result using Perlin's noise function in terms of how natural it looks. One might be able to modulate the noise function by adjusting parameters such that it generates an animated face similar to ours. However, we found that modulating the noise function took significant time and efforts.

Talking baby. Our gesture synthesis method is independent of the language and sentences that the synthetic face talks. The talking baby reads a French story and his gestures are transferred from motion capture data uttering in a different language. It took us only one minute to generate the 20 second talking baby animation.

7. Discussion

The primary advantage of our approach is its capability of reproducing natural variations of human facial gestures in animated faces. In our experiments, we observed the complexity and subtle details of facial gestures that cannot easily be accomplished by using noise functions.

Our system learned expressive facial gestures from a small amount of training data, primarily because we decoupled facial gestures from phonemes and lip movements. Although we have not yet attempted to verify the interrelation between prosodic features and facial gestures, the intensity and pitch of the speech seem to be two major factors strongly synchronized with facial gestures. We made use of the intensity information in our experiments. It would be interesting future work to incorporate the pitch information as well in our system.

All facial models used in our experiments were designed manually by artists and thus the resulting animation looks cartoon-like rather than realistic. There are several guidelines for selecting basis expressions and designing their corresponding face geometry. We used only six basis models (neutral, sad, happy, afraid, angry, surprise) in our experiments. More basis models can be employed for representing a wider variety of facial expressions. We found that it is important to choose a basis expression that exhibits a coordination of facial features across the entire face. Otherwise, it will interfere with stochastic movements of individual facial

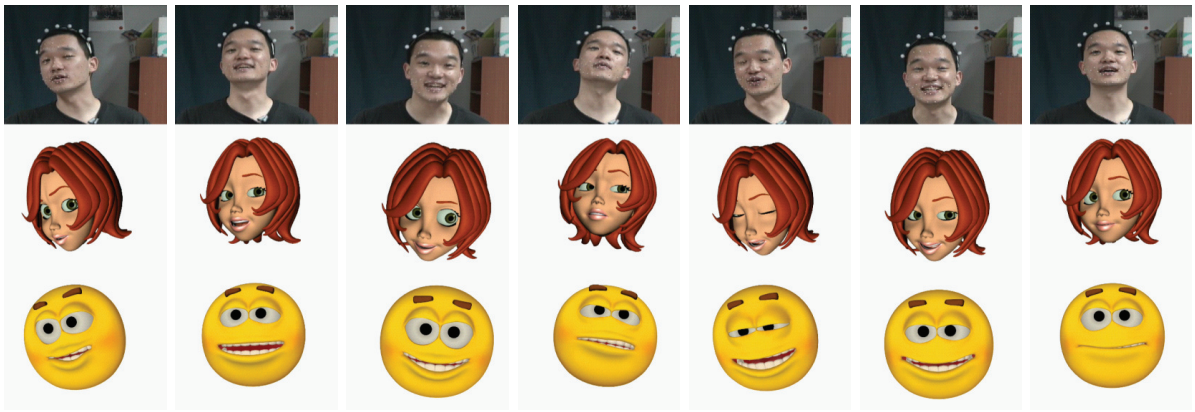

Figure 9: *Facial expression transfer from motion capture data to synthetic faces.*

features and thus the blend weight estimation can be noisy. We prefer to have our face geometry models with a closed mouth. An open-mouth expression would cause mouth opening even for plosives in lip-synching. In order to make more realistic facial animation, our method can still be used with the basis models acquired from 3D face scanning.

We have visually compared the original motion capture data and animated faces to see if natural human gestures are successfully captured and reproduced in the animated faces. Though this visual comparison is an effective way of evaluating the quality of facial gestures in a subjective point of view, we also need a quantitative method for evaluating and characterzing facial gestures.

Acknowledgements

We would like to thank all the members of the SNU Movement Research Laboratory for their help in collecting motion data. This work was partly supported by the Korea Science and Engineering Foundation (R01-2007-000-11560-0).

References

[BB02] BYUN M., BADLER N. I.: FacEMOTE: qualitative parametric modifiers for facial animations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2002), pp. 65–71.

[BCS97] BREGLER C., COVELL M., SLANEY M.: Video rewrite: driving visual speech with audio. In *Proceedings of SIGGRAPH 97* (1997), pp. 353–360.

[BDG*07] BUSSO C., DENG Z., GRIMM M., NEUMANN U., NARAYANAN S.: Rigid head motion in expressive speech animation: Analysis and synthesis. *IEEE Transactions on Audio, Speech and Language Processing 15*, 3 (2007), 1075 – 1086.

[BN07] BUSSO C., NARAYANAN S.: Interrelation between speech and facial gestures in emotional utterances: A single subject study. *IEEE Transactions on Audio, Speech and Language Processing* (2007), In Press.

[Bra99] BRAND M.: Voice puppetry. In *Proceedings of SIGGRAPH 99* (August 1999), pp. 21–28.

[BW07] BOERSMA P., WEENINK D.: Praat: doing phonetics by computer. http://www.praat.org/, 2007.

[CB05] CHUANG E., BREGLER C.: Mood swings: expressive speech animation. *ACM Transactions Graphics 24*, 2 (2005), 331–347.

[CPB*94] CASSELL J., PELACHAUD C., BADLER N., STEEDMAN M., ACHORN B., BECHET T., DOUVILLE B., PREVOST S., STONE M.: Animated conversation: Rule-based generation of facial expression gesture and spoken intonation for multiple converstaional agents. In *Proceedings of SIGGRAPH 94* (July 1994), pp. 413–420.

[CTFP05] CAO Y., TIEN W. C., FALOUTSOS P., PIGHIN F.: Expressive speech-driven facial animation. *ACM Transactions Graphics 24*, 4 (2005), 1283–1302.

[CXH03] CHAI J., XIAO J., HODGINS J.: Vision-based control of 3d facial animation. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), pp. 193–206.

[DN06] DENG Z., NEUMANN U.: efase: expressive facial animation synthesis and editing with phoneme-isomap controls. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), pp. 251–260.

[EGP02] EZZAT T., GEIGER G., POGGIO T.: Trainable videorealistic speech animation. *ACM Transactions on Graphics (SIGGRAPH 2002)* (2002), 388–398.

[EMMT04] EGGES A., MOLET T., MAGNENAT-THALMANN N.: Personalised real-time idle motion

synthesis. In *Proceedings of Pacific Graphics 2004* (2004), pp. 121–130.

[JTDP03] JOSHI P., TIEN W. C., DESBRUN M., PIGHIN F.: Learning controls for blend shape based realistic facial animation. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), pp. 187–192.

[KMT03] KSHIRSAGAR S., MAGNENAT-THALMANN N.: Visyllable based speech animation. *Computer Graphics Forum 22*, 3 (2003), 632–640.

[LCR*02] LEE J., CHAI J., REITSMA P. S. A., HODGINS J. K., POLLARD N. S.: Interactive control of avatars animated with human motion data. *ACM Transactions on Graphics (SIGGRAPH 2002) 21*, 3 (2002), 491–500.

[LTW95] LEE Y., TERZOPOULOS D., WALTERS K.: Realistic modeling for facial animation. In *Proceedings of SIGGRAPH 95* (1995), pp. 55–62.

[MA06] MOUNT D., ARYA S.: Ann: Library for approximate nearest neighbor searching, http://www.cs.sunysb.edu/ algorith/implement/ann/distrib/index1.html, 2006.

[MKPG05] MUELLER P., KALBERER G. A., PROESMANS M., GOOL L. V.: Realistic speech animation based on observed 3d face dynamics. *IEE Proc. Vision, Image & Signal Processing 152* (August 2005), 491–500.

[Per85] PERLIN K.: An image synthesizer. In *Proceedings of SIGGRAPH 85* (1985), pp. 287–296.

[Per95] PERLIN K.: Real time responsive animation with personality. *IEEE Transactions on Visualization and Computer Graphics) 1*, 1 (1995), 5 – 15.

[Per07] PERLIN K.: Improving noise. *ACM Transactions on Graphics (SIGGRAPH 2002) 26*, 3 (2007), 681 – 682.

[PG96] PERLIN K., GOLDBERG A.: Improv: A system for scripting interactive actors in virtual worlds. In *Proceedings of SIGGRAPH 96* (1996), pp. 205–216.

[PKC*03] PYUN H., KIM Y., CHAE W., KANG H. W., SHIN S. Y.: An example-based approach for facial expression cloning. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), pp. 167–176.

[SSRMF06] SIFAKIS E., SELLE A., ROBINSON-MOSHER A., FEDKIW R.: Simulating speech with a physics-based facial muscle model. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), pp. 261–270.

[TO99] TURK G., O'BRIEN J. F.: Shape transformation using variational implicit functions. In *Proceedings of SIGGRAPH 99* (1999), pp. 335–342.

[VBPP05] VLASIC D., BRAND M., PFISTER H., POPOVIĆ J.: Face transfer with multilinear models. *ACM Transactions on Graphics (SIGGRAPH 2005) 24*, 3 (2005), 426–433.

[YRVB98] YEHIA H., RUBIN P., VATIKIOTIS-BATESON E.: Quantitative association of vocal-tract and facial behavior. *Speech Communication 26*, 1-2 (1998), 23 – 43.

[ZLGS03] ZHANG Q., LIU Z., GUO B., SHUM H.: Geometry-driven photorealistic facial expression synthesis. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2003), pp. 177–186.

[ZSCS04] ZHANG L., SNAVELY N., CURLESS B., SEITZ S. M.: Spacetime faces: High-resolution capture for modeling and animation. *ACM Transactions on Graphics (SIGGRAPH 2004) 24*, 3 (2004), 548–558.