MPEG-4 FAP Animation Applied to Humanoid Robot Head

Maher Ben Moussa[1], Zerrin Kasap[1], Nadia Magnenat-Thalmann[1], and David Hanson[2]

[1] MIRALab, University of Geneva
Route de Drize 7
Carouge, Switzerland
{benmoussa,zerrin,thalmann}@miralab.ch
http://www.miralab.ch
[2] Hanson Robotics
1208 W. Executive Drive Richardson
TX 75081, USA
david@hansonrobotics.com
http://www.hansonrobotics.com

Abstract. In this paper, we present a new novel method for creating facial expression for humanoid robots. We map the facial animation parameters (MPEG-4 FAPs) of a virtual face to a robotic face and show how the facial animation of a virtual character can be used to generate expressions on the robotic face.

Keywords: MPEG-4, FAP, Hanson Robotics, Facial Animation

1 Introduction

Robots that look and act like humans have forever captured the imagination of science fiction writers and researchers working in robotics and human interaction alike. In the recent years, there has been a lot of progress in the development of realistic looking humanoid robots. The development of realistic facial expressions for robots has been for a long time one of the important challenges in this field of research. Some researchers have developed toy-like robotic heads such as [14] [12] and other researchers have developed more human-like robotic heads such as [3] [13]. In this paper, we focus on facial expressions and we use a robotic head developed by Hanson Robotics [1] that has extremely realistic looking skin giving it almost a lifelike appearance to capture the mechanics of the face. To animate this robotic head, we relay on knowledge gained from the field of computer graphics. In this field, MPEG-4 Facial Animation Parameters (FAPs) [2] is widely used to animate the facial expressions of virtual humans. In our approach, we are mapping the MPEG-4 FAPs to the movements of the servo-motors that control the facial expressions of the robotic face thus transferring the facial animation from virtual to the robot.

The advantage of using a virtual character as an intermediate stage is that it can be used to design and test the interaction on the software level. The virtual character can be considered as a layer transparent to the user, as the user directly interacts only with the robot. This layered paradigm thus eases the design and experimentation process while making the actual interaction experience between the robot and the human richer and more diverse. Another advantage is that by using MPEG-4 FAP to animate the robotic head, we relay on a widely accepted facial animation standard. This means that different kind of animations from different sources, either produced by designer effort or from motion capture, can be applied to the robotic face. It makes it also possible to integrate any MPEG-4 FAP based face engine to the robot.

In the next section, we will begin with an explanation of the related concepts and previous work that has been done in the area. In Section 2 we describe the MPEG-4 FAPs standard and how it is applied for animation. In Section 3 we give the internal details of the robotic face we use and explain how we animate the robotic face by mapping MPEG-4 FAPs to robot servos in Section 4. Then we continue with explanation of our animation architecture and the model for emotions in Section 5. In Section 6 we present the results of our system. We finally conclude with a short discussion of the limitations of the current system and the directions of future work in Section 7.

2 Animating a face using MPEG-4 FAPs

MPEG-4 FAP is applied as a parameterization method for facial animation. Parameterization, which stands for the manipulation of the parameters from animators' point of view, is one of the important aspects in 3D animations. In the late 1990s, the MPEG-4 group introduced the Facial Animation Parameters (FAP) as a standard set for facial animation on synthetic 3D faces [2]. Since its introduction, it has been successfully adopted by several facial animation systems. As shown in figure 1 , MPEG-4 works by defining several feature points known as Facial Definition Parameters (FDPs) on top of a facial skin mesh and animates those feature points using the corresponding Facial Animation Parameters (FAPs). Each FAP value corresponds to the displacement of one FDP on the face in one direction. FAPs are based on the study of minimal facial actions and are closely related to muscle actions. The FAP values are always relative to key facial distances such as the distance between the eyes or the two lip corners, which are known as Facial Animation Parameter Units (FAPUs). This makes it possible to apply the same FAP parameters on different faces. MPEG-4 defines 68 FAPs that can be animated and they are a subset of the 84 Facial Definition Parameters (FDPs) that can be used to shape the face.

According to our interaction architecture described in Section 5, an expression/viseme database is created based on MPEG-4 FAPs and they are blended in real-time based on the current state of the dialogue and emotional state of the robot. The final blended FAP animation stream is applied on the robotic

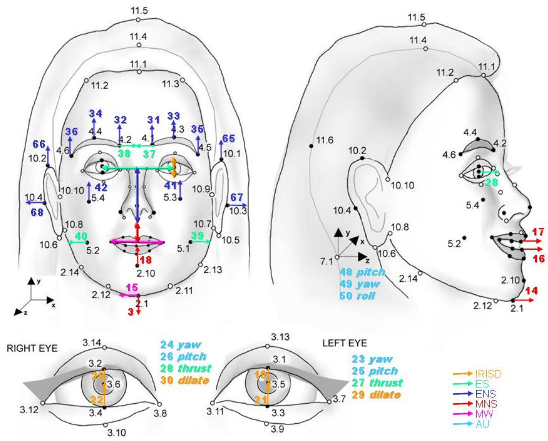

Fig. 1. MPEG4 FDP and FAP

face by the help of the FAP to robot servos conversion algorithm. The following section describes how this algorithm works.

3 THE ROBOTIC FACE

3.1 General details

For the described research, Hanson Robotics created a custom robotic face with 32 degrees of freedom representing the major 48 muscle actions of the human face (Figure 2). The skin of the robot is distinctive, being made from a structured porosity elastomer material developed and patented at Hanson Robotics, and marketed under the brand name Frubber. Being a porous elastomer, this Frubber material requires 20x less force to affect simulated facial expressions relative to solid elastomers [4]and better physically simulates the expressive nature of human facial flesh [3], allowing for smaller motors to generate expressions that are better than can be achieved with other materials. Frubber combined with the Hanson facial mechanisms allow for a full range of expressions, which are naturalistic, human in scale, and real-time. The robot face began as a custom sculpture made by David Hanson, molded using a urethane molding process. The skull is sculpted by referring to forensic facial reconstruction processes, so that the tissue thickness of the skin is anatomically correct in ways that allow for natural looking expressions. The skull form is laser scanned, and the data from this scan is used as the basis of a custom mechanical design, generated in Solidworks CAD software. The resulting mechanism is printed in ABS plastic using a Stratasys 3D printer. The skin is cast into the urethane mold, using the skull-form as the interior, or core, of the mold. Flexible rubber-cloth anchors are designed to distribute force through the skin in a manner that wrinkles, folds, and bunches expressively like real skin. These anchors are cast directly into the

Frubber material during the skin casting process, by precise placement on the core using custom registration pins. Once the Frubber skin is cast, it is treated and attached to the skull, and connected to servo-motors which reside inside the skull.

Fig. 2. Robotic head from front, side and back view

3.2 The details of the robot

The robot has 32 degrees of freedom that simulate the expressive capabilities of face, actuated by reversible servo-motors (Figure 3). Being reversible (versus human muscles, which are unidirectional, not bidirectional), each motor can simulate the action of 2 individually controlled muscle groups in the human face. Thus, the robot is able to simulate the actions of all major muscle groups in the face and neck, including the complex, compounding, coordinated muscle actions responsible for the diversity of human facial expressions, eye motions, and speech-related mouth motions. The fast action of the servos and Frubber skin correlates with the fast response times of human musculature, so enables real-time social interaction with humans. The robot uses Hitec Servomotors (also known as a "servo"), containing a gearhead motor, a position sensor, and a motor control board, contained within a black rectangular box. The servomotor also has a rotational output, which imparts mechanical motion to one degree of freedom of the robot, and a three-wire electric cable that enters the base of the black casing, to provide power and control signal to the servo, in the form of pulse width modulation (PWM), position-commands. These PWM signals are generated by a central controller board, SSC-32. One of the robot's eyes (optionally both eyes) contains a CCD camera (model PC223XP with Sony color imaging elements, providing .2 lux light sensitivity). Each eye is independently actuated on horizontal rotation (eye turning action), enabling stereo-vergence; however the eyes are coupled for vertical rotation (eye up-down action), so move together vertically.

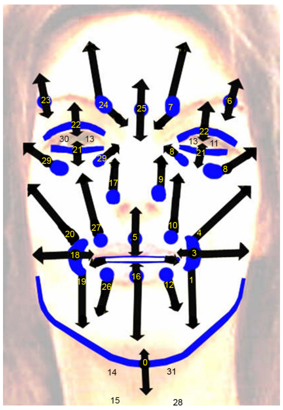

Fig. 3. The servo motors on the robot face and the movements they control

4 ANIMATING THE ROBOTIC FACE

4.1 Mapping

The FAPs are mapped to the movements of the servo-motors that control the facial expressions of the robotic face thus transferring the facial animation from virtual to the robot. The following diagram shows how the mapping is realized:

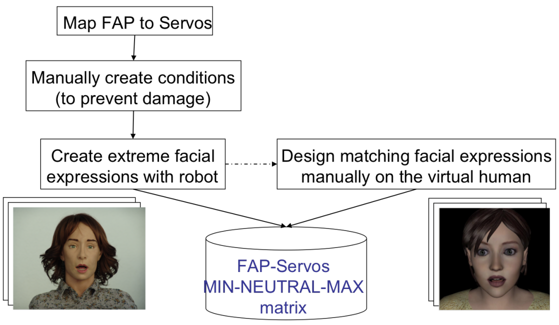

Fig. 4. Mapping MPEG-4 to FAP

To begin with, FAP parameters and FAP ranges should be mapped to corresponding servo motors and servo ranges respectively. This is done in a sequence of three steps.

In the first step, we map FAP parameters to servo motors in the robot. By observing the effects of the movements of each servo on the robot face, a table is created to indicate which FAP parameters are influenced by which servos. We assume that only those FAP parameters that are significantly affected by a servo can be considered as related to that servo.

The previous step results in FAP-Servo conversion rules where each FAP parameter influences one or more servo motors and each servo motor influences one or more FAP parameters. One FAP parameter controlling more than one servo is acceptable, but two FAP parameters controlling the same servo at the same time is not desirable. This is avoided by conditional execution of the conversion rules so that conflicting rules do not execute together. We obtain a FAP-Servo-condition rules database at the end of this step. The FAP system has a different range of values than that of the servo motors in the robot. Therefore we created a conversion between these values. The servos in the robot accept values between 0 and 255. However, because of mechanical limitations, the ranges are often smaller than the servo ranges. For example, ranges of the upper eyelids are limited to 23 and 181.

In the last step, the minimum, maximum and default values of the servos and the FAP parameters are mapped to each other. To determine the minimum and maximum for the FAP parameters, every servo is moved to its extremities and FAP animations were created on a virtual human to match the effect of the servo on the robot face. Those FAP values were considered as minimum and maximum for the FAP. Some servos also had different orientation than the FAP parameters. For those servos, the minimum and maximum are flipped. In order to determine the servo values for the neutral position of the face we used another approach. For FAP parameters, the neutral position of the face corresponds to 0 and therefore the robot face was animated to match the neutral face of a virtual human and the values derived from the servos were considered as the neutral face values for the servo motors.

4.2 The conversion

The mapping resulted in a matrix, which is later used for the conversion. At every frame, the conversion algorithm converts the FAP animation to servo values. The following diagram shows how the conversion works:

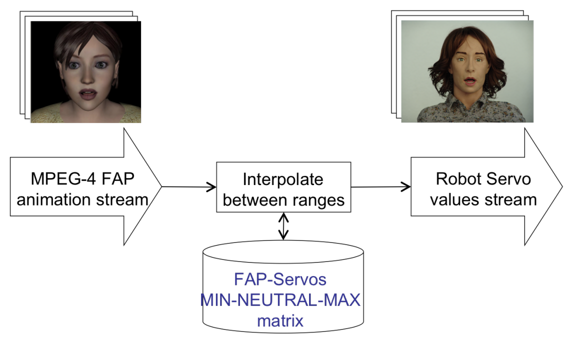

Fig. 5. Conversion of MPEG-4 animations to Robot

In order to convert a FAPs animation to the servo movements we interpolate between the minimum, neutral and the maximum position using cosine interpolation, where the FAP values are considered as X dimension and the servo values are considered as the Y dimension.

5 Integration with an Interaction Framework

In the previous section we presented the FAP to Robot conversion algorithm. In this section we propose an interaction framework integrating this algorithm as shown in figure 6.

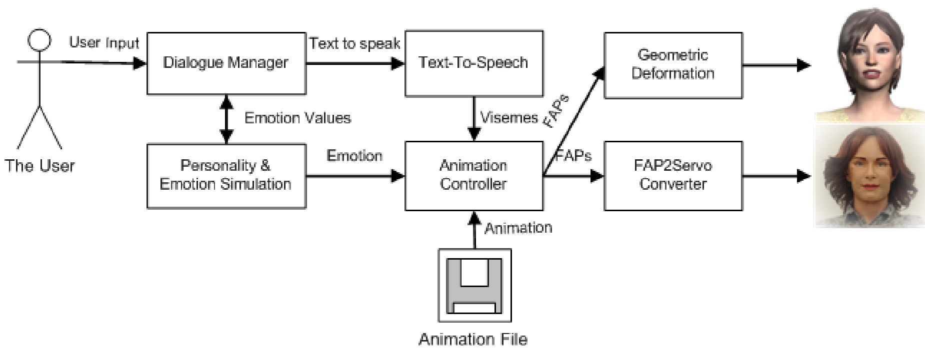

Fig. 6. Interaction Framework Architecture

Our system consists of several modules that are responsible for perceiving the user's input, processing it and generating the correct output. The user interactions with the *Dialogue Management* module, which in its turn analyses the user's input and produces the textual output and the emotion impulse. The textual output is sent to the *Text-To-Speech* module to produce speech output and visemes and the emotion impulse is sent to the *Personality & Emotion Simulation* module to influence the ECA's internal emotions. The *Animation Control* module generates FAP facial animations by combining the visemes and the emotional facial expressions from the modules *Text-To-Speech* and *Personality & Emotion Simulation* and sends the FAP animation to the *Geometric Deformation* module and the *FAP-To-Robot* modules for generating the facial expressions. The *Animation Control* module is also able to load predefined animation sequence from a file and play it.

Next to the *FAP-To-Robot* module that we have presented in the previous section, the following modules are also implemented:

Dialogue Management module is based on Finite-State-Machines (FSMs) that are easy to define and flexible enough to generate a wide variety of output

[5]. Transitions in each FSM consist of conditions and actions that determine the respond based on the user's input and the internal emotional state.

Personality & Emotion Simulation module that we developed maintains the emotional state and updates it dynamically according to the impulses coming from the environment [6]. This module is based on different theories from psychological and represents the affective states as personality, mood and emotions. The personality model is based on the OCEAN model composed of five dimensions: openness, conscientiousness, extraversion, agreeableness and neuroticism) [7]. For emotions, we consider sixteen emotions from Ortony-Clore-Colins (OCC) appraisal model [8], eight being positive (joy, hope, relief, pride, gratitude, love, happy-for, gloating) and eight being negative (distress, fear, disappointment, remorse, anger, sorry-for, resentment). Mood is represented with three dimensions: pleasure, arousal and dominance (PAD) [9] resulting in eight different mood types (exuberant, bored, dependent, disdainful, relaxed, anxious, docile and hostile).

Text-To-Speech module is based on *Microsoft Speech* technology [3] and is responsible for generating the speech output of the virtual human or robot. The module retrieves the textual output from the Dialogue Management module and pronounces it using *Microsoft Text-To-Speech engine*. For lip synchronization, the module obtains the visemes information from Microsoft's engine and passes it to the Animation Control module

Animation Control module generates emotional facial expressions, lip movements and other facial gestures based on information retrieved from other modules and blends them together in order to get realistic real-time facial animation. Lip animation is achieved by generating lip movements corresponding to the visemes retrieved from the Text-To-Speech module. This module is based on Principal Components (PCs) as described by [10] which are derived from the statistical analysis of the facial motion data during fluent speech.

Geometric Deformation module is responsible for rendering the 3D face and deforming it. This engine is controlled by MPEG-4 Facial Animation Parameters (FAP) and is based on feature point based geometric deformation [11]. Depending on the animation values defined as FAPs, the facial deformation engine deforms the facial mesh by displacing each vertex affected according to the displacement values. These displacement values are automatically calculated in the design stage according to the distance between the feature points and the vertices.

[3] http://www.microsoft.com/speech

6 Generating the Facial Animation

We now describe how the affective state update generates the facial animation and expressions for the artificial characters. We show examples on both virtual and robotic faces to illustrate that the same method works for any kind of artificial character when we use the mapping described in Section 4. Based on the emotion engine we have described in the previous section, the expressions are rendered according to two different layers: mood state and emotional state. First, the range of expressions of a character is limited to its mood and indirectly to its personality. For example, in a joyful emotional state, if the character is highly exuberant, she will show a more cheerful expression. Instead, if the character is moderately exuberant, she will have a more controlled joyous expression. Second, each expression also varies in itself, based on the intensity of the current emotional state. If the current emotional state is a positive one and the mood state is also positive, expression is generated based on the intensity of the positive emotion. If mood type and emotional state are conflicting, increase in the intensity of emotional state decreases the intensity of expression. A database of 24 expressions is used for the rendering of emotions, 12 for the positive ones (and) and 12 for the negative ones (and) and for each mood there are three levels of expressions: high, medium and low. Each expression represents the maximum intensity that can occur at a specific mood level. The database is created based on observation of motion capture animations and classification of static expressions based on the pleasure, arousal and dominance values. While pleasure decides the positivity and negativity of the expression, the head motion is related with the dominance component (head up means high dominance and head down means low dominance) and arousal is more related with the energy of the expression. The arousal dimension is also considered during the animation, deciding on how fast each expression is going to appear and disappear.

7 Conclusion

In this paper we illustrated how to map a widely accepted facial animation standard, namely MPEG-4 FAPs to a robotic face. This conversion makes it possible to use animations from different sources, like from motion capture or from a 3D modeling and animation software (e.g. 3ds Max). It is also possible to integrate any MPEG-4 FAP based facial animation engine to the robot. In addition, since our architecture works for any kind of artificial character we can use a virtual character as an intermediate representation. This gives us the additional advantage of being able to test and design the interaction entirely at software level, before we experiment with the robot. This is important because the robot is a delicate and expensive equipment.

Acknowledgments. This study is partly funded by the EU Project 3DLife (IST-FP7 247688).

Fig. 7. Positive and Negative Emotions on Virtual Human and Robot

References

1. Hanson Robotics, `http://www.hansonrobotics.com`
2. Pandzic, I. S. and Forchheimer, R. 2002. MPEG-4 Facial Animation: The Standard, Implementation and Applications. Wiley
3. Hanson, D. Olney, A. Zielke, M. and Pereira A. 2005. Upending the Uncanny Valley. Proc. of Nat'l Conference on Artificial Intelligence (AAAI '05). Pittsburgh, PA
4. Hanson, David, and V. White. 2004. Converging the Capabilities of Electroactive Polymer Artificial Muscles and the Requirements of Bio-Inspired Robotics.
5. E. A., Z. X., K. S., and M.-T. N. Emotional communication with virtual humans. pages 243–263, 2003.
6. Z. Kasap, a. C. P. Ben Moussa, M., and N. Magnenat-Thalmann. Making them remember: Emotional virtual characters with memory. IEEE Computer Graphics and Applications, 29(2):20–29, 2009.
7. R. R. Mccrae and P. John. An introduction to the five-factor model and its applications. Journal of Personality, 60:175–215, 1992.
8. A. Ortony, G. L. Clore, and A. Collins. The Cognitive Structure of Emotions. Cambridge University Press, 1988.
9. A. Mehrabian. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. Current Psychology, 14:261–292, 1996.
10. S. Kshirsagar, T. Molet, and N. Magnenat-Thalmann. Principal components of expressive speech animation. In Proc. Computer Graphics International 2001, pages 38–44. IEEE Publisher, February 2001.

11. S. Garchery, A. Egges, and N. Magnenat-Thalmann. Fast facial animation design for emotional virtual humans. In Proc. Measuring Behaviour, Wageningen, N.L. CD-ROM Proceeding, September 2005.
12. Breazeal, C. 2002. Emotion and sociable humanoid robots. International Journal of Human-Computer Studies 59, 119-155.
13. Nishio S., Ishiguro, H. and Hagita, N. 2007. Geminoid: Teleoperated Android of an Existing Person. In Humanoid Robots: New Developments, de Pina Filho, A.C. eds., I-Tech Education and Publishing, Vienna, Austria
14. Landon, R. 2003. Chapter 6: Character Creation - Structural Elements of Biomimetic Robots / 145, in Biologically Inspired Intelligent Robotics, Ed. Yoseph Bar-Cohen and Cynthia Breazeal, SPIE Press