Virtual Emotion to Expression: A Comprehensive Dynamic Emotion Model to Facial Expression Generation using the MPEG-4 Standard

Paula Rodrigues
Informatics Department, PUC-Rio, Brazil
paula@inf.puc-rio.br

Asla Sá
TecGraf, PUC-Rio, Brazil
aslasa@gmail.com

Luiz Velho
IMPA - Instituto de Matemática Pura e Aplicada, Brazil
lvelho@impa.br

Abstract

In this paper we present a framework for generating dynamic facial expressions synchronized with speech, rendered using a tridimensional realistic face. Dynamic facial expressions are those temporal-based facial expressions semantically related with emotions, speech and affective inputs that can modify a facial animation behavior.

The framework is composed by an emotion model for speech virtual actors, named VeeM (Virtual emotion-to-expression Model), which is based on a revision of the emotional wheel of Plutchik model. The VeeM introduces the emotional hypercube concept in the R^4 canonical space to combine pure emotions and create new derived emotions.

The VeeM model implementation uses the MPEG-4 face standard through a innovative tool named DynaFeX (Dynamic Facial eXpression). The DynaFeX is an authoring and player facial animation tool, where a speech processing is realized to allow the phoneme and viseme synchronization. The tool allows both the definition and refinement of emotions for each frame, or group of frames, as the facial animation edition using a high-level approach based on animation scripts. The tool player controls the animation presentation synchronizing the speech and emotional features with the virtual character performance. Finally, DynaFeX is built over a tridimensional polygonal mesh, compliant with MPEG-4 facial animation standard, what favors tool interoperability with other facial animation systems.

Keywords: Facial Animation, Talking Heads, Expressive Virtual Characters.

1 Introduction

Character Animation is one of the key research areas in Computer Graphics and Multimedia. It has applications in many fields, ranging from Entertainment, Games, Virtual Presence and others.

Within the general area of character animation, the modeling and animation of faces is perhaps the single most important and challenging topic. This is because the expressiveness and personality of a character is communicated by facial expressions.

The research in face modeling and animation dates back to the seminal work of Frederic Parke in the early 1970's [9]. Since that time, the area experienced a very intense development. Practically all problems related to generating the shape and motion of faces have been deeply studied. This body of research includes a plethora of techniques for capturing the geometry and appearance of human faces, learning facial expressions, modeling muscles and the dynamics of deformations, together with realistic rendering methods for skin and hair.

Despite of the amazing progress in the area of facial animation, there is one problem which is still open, and poses a great challenge to researchers: it is how to incorporate emotion on animated characters! This is the crucial step toward believable virtual characters.

While a talented artist with the help of powerful modeling and animation tools can manually create a very expressive character, the same is not true for an automatic or even semi-automatic animation system.

It is our intent in this paper to address the challenge of generating believable virtual characters automatically by incorporating a computational emotion model. We propose a comprehensive emotion model for facial animation that considers the various aspects of an expressive character. The model is implemented using a system based on the guidelines of the MPEG-4 standard for faces.

The rest of the paper is structured as follows: in next section the emotion models and related work are discussed. In Section 3 we propose an emotion space named *emotion Hypercube*. The emotion hypercube enables pure emotion combinations in order to generate derived emotions in a natural way. We then describe a derived emotion classification scheme based on the proposed space. In Section 4 some affective phenomenon, like mood and personalitity, are incorporated into the model. In Section 5 the *VeeM* (*Virtual emotion-to-expression Model*) is formalized, it consists of a representation of a facial expression from an emotion description with dynamic features. Section 6 presents an overview of the MPEG-4 standard to facial animation. In Section 7 we explain how the proposed model is implemented. Finally, conclusions and future work are discussed in the Section 8.

2 Emotion Models and Related Work

Several models have been proposed to explain what is an emotion and how it is represented [13]. Here we summarize the main approaches related to this topic.

Basic Emotion is probably the most well-known emotion approach. The reason for this is its association with universal recognized emotions [5]. Nevertheless, there is not a consensus for defining which are the basic emotions yet.

As discussed in [7], the basic emotion approach aims to build a *psychologically irreducible* emotion set, which means that these emotions cannot be derived by any other emotion and new emotions are derived from them. Note that these considerations matches to the mathematical definition of a basis.

As mentioned above, the best known method used to study basic emotion is by observing facial expressions. Through this, Ekman [5] defined six universal emotions: *anger, fear, disgust, surprise, joy* and *sadness*, illustrated in Figure 1.

An extension of Ekman's model to the basic emotions representation is

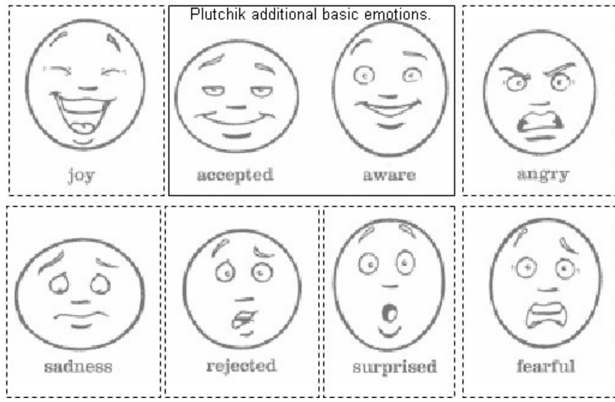

Figure 1: Six universal basic emotions defined by Ekman (surrounded by a dashed line) and additional Plutchik basic emotions (*accepted* and *aware*).

the approach proposed by Plutchik [11], where two additional basic emotions are defined (emphasized in Figure 1): *anticipation* (also referred as *aware, curiosity* or *interest*) and *acceptance* (also referred as *trust*). Plutchik describe its basic emotion as pairs of opposite emotions.

Plutchik emotions are disposed in a wheel of opposed pairs, as illustrated in Figure 2. Derived emotions are defined as the combination of two neighbor basic emotion or as a basic emotion intensity variation. In the emotion literature, the Plutchik wheel is considered to be enough to span most of human emotion state.

Examples of computational systems that use the basic emotion approach to generate their facial expressions are: SMILE [6], eFASE [3], EE-FAS [15], Cloning Expression [12], the MPEG-4 Standard [8] and the CSLU Toolkit [2].

In addition to any model of emotion, the emotion perception becomes single for each person due to factors such as mood and personality.

The approach proposed in [10] is to model mood as a simple and unique dimension: *good* mood and *bad* mood. A more complete approach proposed by Thayer in [16] uses emotion spaces to represent mood in two dimensions (*calm/tense* and *energy/tired*), resulting in four mood emotional states: *Energetic-calm, Energetic-tense, Tired-calm* and *Tired-tense*. An example of computational system that incorporates mood to the character using Thayer's model to generate dynamic facial expressions is the DER [14] [15].

Personality is another important aspect to define the action and reaction of each person as unique even when submitted to the same situation

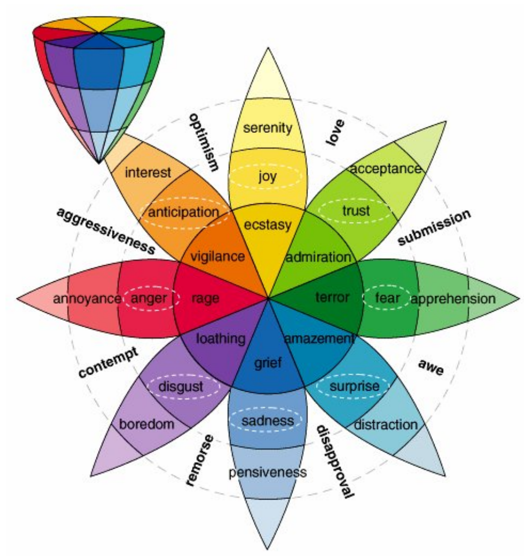

Figure 2: Plutchik wheel.

of another person. Until now, there is not a formal consensus to define the personality trait of a person, however the *Big Five* (or Big OCEAN) model is well-known. In this model, each first letter of OCEAN word defines a dimension in the personality trait: **O***peness to experience,* **C***onscientiousness,* **E***xtraversion,* **A***greeableness,* **N***euroticism* [1].

Emotions are not static. They are experienced by each individual differently because of characteristics such as personality and mood, referred here as *affective phenomenas*. Additionally, affective phenomena also interferes in the reaction of each person when receives a stimulus, defining the *emotion sustaining time*.

Our aim is to propose a computational system which incorporates and implements a robust model based on basic emotions. So, the Plutchik model [11] is revisited and generalized to allow the description of new emotions from the eight basic emotions as well as the incorporation of emotion dynamics in a comprehensive manner in order to allow the automatic generation of believable virtual characters.

[1]More information about BIG OCEAN model can be found at *http://www.answers.com/topic/big-five-personality-traits* (accessed in 22-jan-2008).

3 The Emotion Hypercube

The emotion description space proposed in this paper is a reinterpretation of Plutchik's emotional wheel.

We consider a *family of emotions* as a set of emotions composed by different intensity levels of a given basic emotion E_i. Plutchik has considered a discrete set of three levels of intensity of a given basic emotion, namely, an attenuation of the pure basic emotion, the pure basic emotion itself and an extrapolation of the pure basic emotion, as described in Table 1.

Table 1: Basic emotions

family (axis)	attenuation ($\|\alpha_i\| < 1$)	basic emotion ($\|\alpha_i\| = 1$)	extrapolation ($\|\alpha_i\| > 1$)
1 (x+)	serenity	**joy**	ecstasy
2 (y+)	annoyance	**anger**	rage
3 (z+)	acceptance	**trust**	admiration
4 (w+)	distraction	**surprise**	amazement
5 (x-)	pensivenes	**sadness**	grief
6 (y-)	apprehension	**fear**	terror
7 (z-)	boredom	**disgust**	loathing
8 (w-)	interest	**anticipation**	vigilance

Assuming that the Plutchik's set of basic emotions is psychologically irreducible, our goal is to define a basis that represents the space of derived emotions. In order to do so, we define an *emotion axis*, denoted by e, as composed by a pair of opposed families as stated in Plutchik emotional wheel. Thus, the eight basic emotions are arranged in 4 emotion axes denoted by x, y, z e w.

The level of intensity of an emotion axis is modeled as a continuum parameter represented by the real value α_i, where $\alpha_e \in [-\gamma, +\gamma]$, with $|\gamma| \geq 1$. A basic emotion is mapped to the intensity level 1 and its opposed basic emotion is mapped to -1. The neutral emotion is mapped to level 0. Though the intensity level $|\alpha_i| = 1$ matches to a basic emotion, $|\alpha_i| < 1$ is the emotion attenuation and $|\alpha_i| > 1$ is the emotion extrapolation (See Table 1).

The more obvious space to be adopted to represent the space of emotions is R^n, where n is the number of opposed pairs of emotions to be considered in the model. Since we adopt 4 emotion axes, limited to the interval $[-\gamma, +\gamma]$

we obtain an *emotion hypercube*

$$\mathcal{H} = [-\gamma, +\gamma] \times [-\gamma, +\gamma] \times [-\gamma, +\gamma] \times [-\gamma, +\gamma]$$

A given emotional stimulus can then be completely defined by a vector of intensity levels $\vec{u} = (\alpha_x, \alpha_y, \alpha_z, \alpha_w)$ that represents the character's state of emotion.

The emotion hypercube $\mathcal{H}$ is a comprehensive emotion space useful to facial expression generation. Observe that the proposed space can be easily extended to more than four axes if a new pair of opposed basic emotions is incorporated, observing that the new axis should be independent from the previously defined ones in order to preserve the property that the set of emotion axes is a basis to $\mathcal{H}$.

3.1 Derived Emotions

The combination of basic emotions is an adequate approach to use with *virtual talking heads*. Plutchik [11] states that two basic emotions can be combined if they are not opposed to each other.

The emotion hypercube $\mathcal{H}$ leads to a simple way to derive combined emotions. For instance, binary combinations can be defined by setting two intensity levels to zero. Thus, given two non opposite basic emotions E_i and E_j, their combination is defined by their non zero intensity values α_i and α_j. Although the original Plutchik model restricts combinations to adjacent basic emotions, we do not adopt this restriction in the proposed model.

We emphasize that by using the hypercube model, n-ary combinations, where n is greater than two, are easily stated, although in the literature combinations of more than two emotions are not considered, probably due to the combinatorial growth of the number of derived emotions to be considered and semantically interpreted. Observe also that the restriction to not combine opposite emotions is intrinsic to the disposition of opposite emotions in the same axis.

3.2 Binary Derived Emotions Taxonomy

In this section we propose a natural and complete taxonomy of the binary derived emotions that extends Plutchik's work [11] by defining additional semantic interpretations.

Two emotion axes e_i and e_j define an emotion plane of derived emotions Π_{ij}. We will refer to each quadrant of a plane as a *sector of derived emotion*. The combination of the 4 axis, 2 by 2, results in 6 derived planes. In Table 2 each sector of each plane is named with a semantic interpretation of the derived emotion.

We emphasize that the semantic interpretation of each sector corresponds to the combination of two basic emotions with level of intensity $|\alpha_i| = 1$. If the basic emotion intensity is attenuated or extrapolated the semantic interpretation of the derived emotion may change.

4 Modeling Affective Phenomena in $\mathcal{H}$

Personality, mood, physical environment and others factors interfere on the expression of emotion. We refer to such factors collectively as *affective phenomena* and when applied to an individual character as its *affective pattern*. Thanks to affective phenomena people under the same emotion stimulus react and feel it in different ways and intensities according to its affective pattern.

Affective phenomena are more enduring emotions and usually occur as an emotional background of much lower intensity than emotional episodes. In order to model the influence of an affective phenomena on an emotional episode we assume that the basic emotions parameters are defined considering a neutral character while an affective pattern is modeled as a distortion (warping) of the original emotion space.

4.1 Affective Pattern Description

Suppose that in a given instant a happiness stimulus is augmented, then it is reasonable to think that the level of intensity related to joy also augments even if the affective pattern of the character is biased to sadness.

Thus it is reasonable to state that an affective pattern can be defined by a set of monotonic functions $f_i : [-\gamma, \gamma] \rightarrow [-\gamma, \gamma]$. The new intensity level on each emotion axis is $\tilde{\alpha_i} = f_i(\alpha_i)$, corresponding to the original stimulus i with $i = x, y, z, w$ is one of the emotion axes from $\mathcal{H}$. Then, the vector $\vec{\tilde{u}} = (\tilde{\alpha_x}, \tilde{\alpha_y}, \tilde{\alpha_z}, \tilde{\alpha_w})$ is an instance of the affective pattern in $\mathcal{H}$.

This approach to modeling affective patterns is general and capable to describe non-neutral characters behavior. The set of functions $\chi = \{f_x, f_y, f_z, f_w\}$

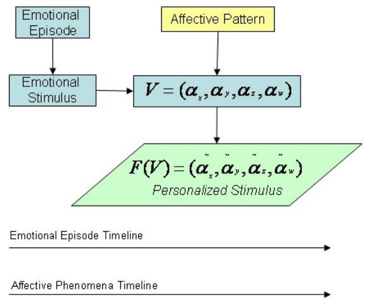

Figure 3: Emotion vector generation.

characterize an affective pattern. The difficulty resides in determine such functions in a meaningful manner.

Note that the physical environment can also be described by a set of such functions. For example if a character is in a formal environment the emotion expression tends to be attenuated, that is, $\tilde{\alpha}_i < \alpha_i$, thus the χ set should be a set of emotion attenuation functions.

4.2 The Dynamics of an Affective Pattern

It is important to notice that affective patterns are not static. For instance, personality traits evolve in lifetime while mood can change during the day time or as a reaction to an emotional episode.

The emotion dynamics can be modeled as a function of emotion and time $f : (\mathcal{H}, t) \rightarrow \mathcal{H}$.

Semantic aspects also need to be taken into account in the context of emotion dynamics. For example: a character cannot mix opposite emotion families and cannot swap them frequently, otherwise a conflict or emotional instability is established. The combination of basic emotions with the farthest families generates destructive emotions, because they are close to a conflict state.

5 VeeM: Virtual Emotion to Expression Model

The $\mathcal{H}$ space is used to define a given character's state of emotion in an instant of time. In order to simulate a believable character animation, the affective patterns and its dynamic characteristics as well as dynamic characteristics

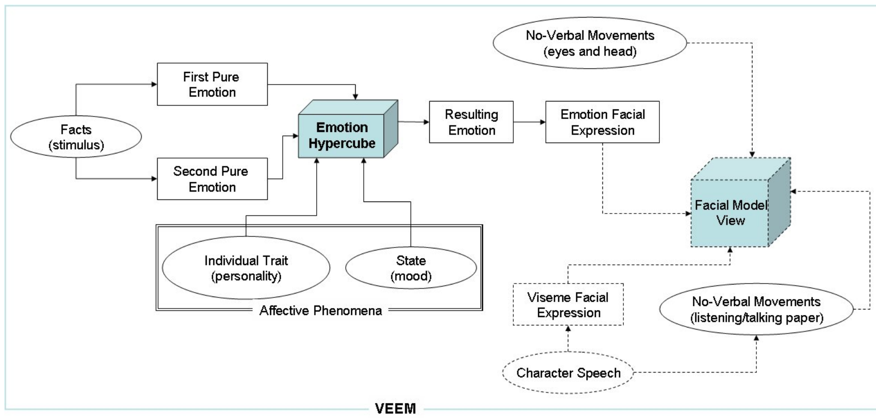

Figure 4: *VeeM* architecture.

such as head movements and eyes blink have to be combined together. The dynamics of the facial expression of a given emotion will be treated by *VeeM*. In Figure 4 a schematic view of the proposed model is illustrated.

There are other equally important dynamic characteristics to be considered when aiming to generate a believable character, namely the *speech* dynamics, the *eyes movements* dynamics and the *head movement* dynamics, refered in literature as *non-verbal movements*. The difference between affective patterns and non-verbal movements is their domain of action, instead of affect an emotional state, the non-verbal movements act directly on the facial expression domain $\mathcal{F}$. Thus, can be modeled as a function of emotion and time $g : (\mathcal{H}, t) \rightarrow \mathcal{F}$.

Speech interferes on mouth movements and a lot of work has been done in order to define the movements that characterize it. The subject is complex and depends on the spoken language and character's culture. But the consensus is to define *visemes* to represent the phonemes and combine them to produce speech visualization. A viseme is a visual representation of a phoneme that describes the facial movements that occur alongside the voicing of phonemes.

Head and eyes movements can be treated as random functions (random noise) considering the fact that it is uncommon to keep them fixed. Other complementary approach is to model them as directed reactions that simulate the attentional focus. In both cases the function that model these behaviors should interfere in the head and eyes position as well as the motion velocity.

Speech and non-verbal characteristics such as eyes and head movements should be combined with the emotion description in order to produce the resultant facial expression [13].

VeeM also incorporates *dynamics of reaction* to an emotion episode (directly related to stimulus or facts which elicits an emotion). The emotion expression reaction depends on the affective pattern. The reaction curve $r_j(t)$, similarly to the description of emotional stimulus given by Picard [10], has 3 stages: *onset* (usually very fast), *sustain* and *decay* stages, as illustrated in Figure 5.

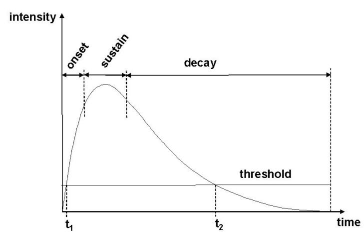

Figure 5: Emotion reaction curve.

Emotion transition is incorporated as a blending between two subsequent emotions.

6 The MPEG-4 Standard

Once the domain of emotion description $\mathcal{H}$ has already been defined we now turn to the description of the range space, that is, the space where the emotions are to be visualized: the facial expression space $\mathcal{F}$.

In general, different works in facial animation generation develop their own facial model without worrying about compatibility. Often the only common approach among these works is the emotion model, which usually are the six Ekman's basic emotions.

Aiming to define a standard, the MPEG-4 [4] [8] agreed a set of control points to define a facial model proposing a facial polygonal mesh that can be considered universal. It is important to mention that the MPEG-4 facial animation standard is the first effort in this direction.

The MPEG-4 specifies a face model in its neutral state, a number of *feature points* (FPs) on this neutral face as reference points, and a set of *facial animation parameters* (FAPs), each corresponding to a particular facial action that deforms the face model starting from the neutral state. In this work we identify the facial expression space $\mathcal{F}$ to the space of FAPs.

A neutral face in the MPEG-4 standard must consider the following properties:

- Gaze is in the direction of z-axes;
- All face muscles are relaxed;
- Eyelids are tangent to the iris;
- The pupil is one third the diameter of the iris;
- Lips are in contact; the line of the lips is horizontal and the same height at lip corners;
- The mouth is closed and the upper teeth touch the lower ones; and
- The tongue is flat, horizontal with the tip of the tongue touching the boundary between upper and lower teeth.

In order to define FAPs for arbitrary face models, MPEG-4 defines facial animation units (FAPUs) that servers to scale FAPs for any face model. FAPUs are defined as fractions of distances between key facial features. These features, such as eye separation are defined on a face model that is in the neutral state.

From the FAPUs definition, MPEG-4 specifies 84 FPs on the neutral face. The main purpose of these FPs is to provide spatial reference for defining FAPs. FPs are arranged in groups such as cheeks, eyes and mouth. The location of these FPs has to be known for any MPEG-4 compliant face model.

The FAPs are based on the study of minimal perceptible actions and are closely related to muscle actions [8]. The 68 parameters are categorized into 10 groups related to parts of the face (Table 3). FAPs represent a complete set of basic facial actions including head motion, eye and mouth control.

The FAP group 1 contains two high-level parameters: visemes and expressions. The MPEG-4 standard defines 14 visemes to represent english phonemes [13] [8]. The expression parameter defines the six basic facial expressions. Facial expressions are animated by a value defining the excitation of the expression. A benefit of using FAP Group 1 is that each facial model preserves its personality in a sense that a specific face model preserves its particular version of facial expression.

FAP groups 2 to 10 are considered low-level parameters. They specify precisely how much a FP of a face has to be moved for a given amplitude [8].

An MPEG-4 facial expression is then obtained by moving the Feature Points (FP) associated to the FAPs. Each basic emotion has a set of FAPs defined to produce its correspondent facial expression. We call signal the facial expression related to a specific emotion and we denote the set of FAP values that define the emotion j as $\vec{v}_j$.

The facial animation sequence is obtained by specifying FAP values at each time instant, $\vec{v}_j^t$, according to an input timeline. We adopt the MPEG-4 Standard as our space of facial expressions.

7 VeeM applied on an MPEG-4 Face Model

As already mentioned, the VeeM permits to generate different facial emotions taking into account affective patterns and the emotion dynamics. A challenge turns into emotion visualization using an MPEG-4 face model. That is, to set FAPs parameter values from the defined emotion.

Figure 6: VeeM basic emotions viewed in a MPEG-4 facial animation.

In this work we adopt the open source *Xface* [1] face model. The model had to be extended since the original implementation defines only the six Ekman's basic emotions as conventioned in MPEG-4 standard. Figure 6 shows the eight basic emotions of VeeM specified in the Xface MPEG-4 polygonal mesh.

Speech is a key element to generate a dynamic and natural facial animation combined with the character's emotional state. The implemented system uses an audio file with the character's speech as input. This file goes through a speech recognition stage generating, as output, the speech phonemes [13]. The produced phonemes are mapped into the 14 MPEG-4 visemes, each one mapped into a set with 26 FAPs that define the mouth region.

The FAPs values generated at each stage of facial animation specification (verbal expressions, non-verbal expressions and emotions) need to be blended to define the final value that a FAP takes in each animation frame. For each animation frame, this blending is done in two stages (Figure 7):

- Facial expression for the emotion (pure or derived); and
- Facial expression for the resulting emotion and viseme blending.

Figure 7: FAP blending to generate the final facial expression for each animation frame.

The first stage can receive as input a single vector of FAPs, two vectors of FAPs or nothing. Whatever the input, the results are always two vectors with dimension 68: a vector containing the mask of FAPs (signaling if a FAP participates or not in the current frame) and another with the values for the participants FAPs.

If a vector is not provided as input, the first stage result is the FAPs vector for natural emotion and a mask containing the value 0 for the 26 FAPs of the mouth region and for the FAP 1 (viseme), and the value 1 for the other FAPs. This configuration purpose is the face remains in the natural state, and its expression influenced only by the viseme FAPs.

In the case where a single FAPs vector is provided, the first stage output is the FAPs vector itself and a mask containing a value of 1 for all FAPs, with the exception of FAP 1 (viseme).

When the first stage receives two vectors of FAPs, a blending is necessary to define the derived emotion. This blending is obtained by calculating the mean between the values of each FAP, except for FAPs 1 and 2 which are the high-level FAPs. In FAP 1 the value 0 is set, once it represents the viseme.

In FAP 2, which describes the emotion, the value 0 is assigned, because this FAP does not has influence in the animation, since only low-level FAPs are considered by the module of synchronization.

If the two FAPs vectors are provided as input, the output mask passes through the same process as in the case of only one input vector.

The second blending stage generates the resulting FAPs vector using as input the first stage output, the viseme FAPs vector and a viseme contribution parameter indicating a factor blending, denoted by β, where $0 \leq \beta \leq 1$.

Before applying the viseme-emotion blending rule, the second stage creates a mask for the viseme FAPs vector.

If there is a viseme FAPs vector, the generated mask vector has value 1 for the 26 FAPs related to mouth region and value 0 for the other FAPs. If there is not a viseme FAPs vector specified as input of second stage, the mask is created with all elements of the vector having value 0.

The blending rule is simple. Denoting FAP_{emo} and $MASK_{emo}$, respectively, the resulting emotion FAPs vector and mask vector; FAP_{vis} and $MASK_{vis}$, respectively, the viseme FAPs vector and mask, both created in second stage; and β as the viseme contribution factor, it is possible to apply for each index i of resulting FAPs vector FAP_{res} the following algorithmic logic:

- if $(MASK_{vis} = 0) \Rightarrow FAP_{res_i} = MASK_{emo_i} * FAP_{emo_i}$
- if $((MASK_{vis} \neq 0) \wedge (MASK_{emo} = 0)) \Rightarrow FAP_{res_i} = MASK_{vis_i} * FAP_{vis_i}$
- if $((MASK_{vis} \neq 0) \wedge (MASK_{emo} \neq 0)) \Rightarrow FAP_{res_i} = (1 - \beta) * MASK_{emo_i} * FAP_{emo_i} + \beta * MASK_{vis_i} * FAP_{vis_i}$

With this rule, it is possible to note that the value $\beta = 1$ is related to the blending visemes and emotions strategy where visemes is overlapping emotions, ignoring the emotion influence in the generation of facial expression in mouth region. The β value can be set for a qualitative analysis of the visual results obtained. Good results have been achieved with values ranging from $0.6 \leq \beta \leq 0.7$.

Once FAP values for each animation frame are generated, the next step is the synchronization between the audio speech file and each frame. In the beginning of animation presentation, a thread for audio presentation is initiated in parallel with the thread used to control the face. The thread responsible for the face control synchronizes FAPs with the audio verifying the machine clock and using the frame frequency defined in the animation. At each iteration, the elapsed time from the start of the animation is calculated and the corresponding frame is loaded. If the frame is different from the previous one, it is taken in FAP structure received as a parameter. This method generates the FAPs mapping in the mesh and the new expression is rendered.

8 Conclusion

This paper introduced a new emotion model for the generation of facial expressions in virtual characters. The proposed *V*irtual *e*motion-to-*e*xpression *M*odel (*VeeM*) is based on a generalization of Plutchik's emotional wheel [11] to the emotional Hypercube $\mathcal{H} \subset R^4$. This mathematical formulation allows the combination of the 8 pure emotions in a general and coherent way. Furthermore, it sets the ground for a comprehensive framework which integrates emotions and affective phenomena by time-varying functions $f : (\mathcal{H}, t) \to \mathcal{H}$ as well as the non-verbal movements $g : (\mathcal{H}, t) \to \mathcal{F}$, from the configuration space $\mathcal{H}$ to the space of facial expressions $\mathcal{F}$. The dynamics of expressions is modeled by considering the temporal properties of functions f_t in the space $\mathcal{H}$ and g_t in the space $\mathcal{F}$.

Facial expressions are defined in our framework using the MPEG-4 standard. Consequently, another relevant contribution of this paper is a computational methodology that incorporates VeeM expressions into the representation of a face under the MPEG-4 guidelines. Our system generates animations of believable virtual characters from emotional and verbal elements. This is done by mapping emotion and speech to FAPs at each frame of the animation with lip-sync and expression dynamics.

An important aspect that we plan to address in the future is the development of tests to validate the combination between visemes and facial expressions in VeeM. Another avenue for future work is the investigation of derived emotions resulting from the combination of more than two axis of the emotional Hypercube.

Further research for the VeeM framework includes: a detailed analysis of warpings of the emotional hypercube to model affective phenomena; an in-depth study of how to effectively model the dynamics of expressions through time-dependent properties of functions in the emotional space; and a validation of the best strategies to perform transition between emotional states. Finally, our end goal is to fully exploit the potential of VeeM in actual applications.

References

[1] Balci, K.: Xface: MPEG-4 based open source toolkit for 3D Facial Animation. Proceedings of the working conference on Advanced visual interfaces, 399–402 (2004).

[2] Cole, R.: Tools for research and education in speech science. Proceedings of the International Conference of Phonetic Sciences (1999).

[3] Deng, Z., Neumann, U.: eFASE: Expressive Facial Animation Synthesis and Editing with Phoneme-Isomap Control. Proceedings of ACM SIGGRAPH/Eurographics Symposium on Computer Animation (2006).

[4] Ebragimi, T., Pereira, F.: The MPEG-4 Book. Prentice Hall PTR, 1st edition (2002).

[5] Ekman, P.: Universal and cultural differences in facial expressions of emotion. Nebraska Symposium on Motivation. Ed. Lincoln, 207–283 (1971).

[6] Karla, P. et al.: SMILE: A Multilayered Facial Animation System. Proceedings of IFIP WG **5 (10)**, 189–198 (1991).

[7] Ortony, A., Turner, T.J.: What's basic about basic emotion? American Psychological Association, Inc. **97 (3)**, 315–321 (1990).

[8] Pandzic, I.S., Forchheimer, R.: MPEG-4 Facial Animation: The Standard, Implementation and Applications. John Wiley and Sons, Ltd (2002).

[9] Parke, F.: A parametric model for human faces. PhD Thesis, University of Utah (1974).

[10] Picard, R. W.: Affective computing. Cambridge, Mass. : M.I.T. Press (1997).

[11] Plutchik, R.: A general psychoevolutionary theory of emotion. Emotion: Theory, research, and experience. Theories of emotion **Vol 1**, 3–33 (1980).

[12] Pyun, H. and et al.: An Example-Based Approach for Facial Expression Cloning. Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, 167–176 (2003).

[13] Rodrigues, P.S.L.: A System for Generating Dynamic Facial Expressions in 3D Facial Animation with Speech Processing. PhD Thesis, PUC-Rio, Brazil (2007).

[14] Tanguy, E. A. R.: Emotions: the Art of Communication Applied to Virtual Actors. PhD Thesis, Department of Computer Science; University of Bath; CSBU-2006-06 (ISSN 1740-9497) (2006).

[15] Tanguy, E., Willis, P., Bryson, J.: A Dynamic Emotion Representation Model Within a Facial Animation System. Techinical Report, CSBU-2005-14 (2005).

[16] Thayer, R.E.: The origin of everyday moods. Oxford University Press (1996).

Table 2: Taxonomy

Derived Plane	basic emotion i	basic emotion j	derived emotion
Π_{xy}	Joy	Fear	thrill
	Joy	Anger	negative pride
	Sadness	Fear	despair
	Sadness	Anger	envy
Π_{xz}	Joy	Trust	love
	Joy	Disgust	morbidness
	Sadness	Trust	sentimentalism
	Sadness	Disgust	remorse
Π_{xw}	Joy	Anticipation	optimism
	Joy	Surprise	absent
	Sadness	Anticipation	pessimism
	Sadness	Surprise	disappointment
Π_{yz}	Fear	Trust	submission
	Fear	Disgust	distress
	Anger	Trust	dominance
	Anger	Disgust	contempt
Π_{yw}	Fear	Anticipation	anxiety
	Fear	Surprise	awe
	Anger	Anticipation	aggression
	Anger	Surprise	outrage
Π_{zw}	Trust	Anticipation	positive pride
	Trust	Surprise	curiosity
	Disgust	Anticipation	cynicism
	Disgust	Surprise	defeat

Table 3: FAPs groups

Group	Number of FAPs
1. visemes and expressions	2
2. jaw, chin, inner lowerlip cornerlips, midlip	16
3. eyeballs, pupils, eyelids	12
4. eyebrow	8
5. cheeks	4
6. tongue	5
7. head rotation	3
8. outer-lip positions	10
9. nose	4
10. ears	4