Growing on the Inside: Soulful Characters for Video Games

Rafael Bidarra, Robert Schaap and Kim Goossens

Abstract— Procedural generation techniques effectively help to reduce the huge content production costs in current video games. In a similar way, we propose to combine a rich internal state model with procedural facial expression generation as a flexible and natural step to solve the proverbial lack of expressiveness in game characters.

I. Introduction

Despite much progress on modeling and graphics, facial expressions of video game characters mostly come out very much canned. Even worse, whatever the actual course of gameplay events, all too often these seem to have little to no effect on the mood, emotions or affective state of the player. At times, the overall level of gameplay experience doesn't go beyond that of mediocre adventure movies, where the hero inexhaustibly defeats everything and everyone on neat garments and impassible features.

Long since, generating captivating facial, and more in general body, expressions for video game characters has proved immensely challenging. The combination of scripting with key frame and/or motion capture animations has often been pointed as an adequate technique to approach this, basically prescribing which character reactions should be triggered, and the corresponding animations performed, as a result of which events, possibly with some fuzziness and/or noise to add some variation; e.g. if player scores a goal, then perform jumps of happiness; if player is hit, then whine or reel…

This worked satisfactorily as long as game storylines were simple, and designers could basically anticipate how to react to each possible event. However, it is easy to see that this approach does not scale. In most current games, their complexity, extension and open-ended storyline simply make it impracticable to anticipate when, or even whether, events will take place. Even worse, the sequence of events is unpredictable, and so is also the successive combination of built-in character reactions and animations, potentially yielding rather unlikely and odd behaviors.

Imagine a multiplayer game situation where a team of soldiers takes part in a war. Their objective is to free a group of hostages. Now consider two different versions of this fight. In both versions of the story our heroes save the hostages, the big difference being the road to their victory. In the first version of the story they meet a lot of resistance, but they manage to free the prisoners without casualties of their own. In the second version one of the team mates does not survive the fight. In your typically scripted computer game, the emotions displayed upon freeing the hostages would be typically the same for both versions of the gameplay: for example, the heroes would raise their arms to the sky and cheer victory. With the approach described in this paper, the journey of our heroes will be taken into account. In the version where they lose one of their brothers in arms, their reactions during the game will reveal despair, sadness; in turn, this will even have an impact on their game-play performance and, eventually, their victory celebration will be fitter and more moderate.

We firmly believe that any solid attempt at solving the problems that the current generation of video games faces will have to satisfactorily approach, at least, the following three main objectives. First, it should empower game designers and artists to declaratively and easily express their creative intent concerning the interior richness of their characters. Second, it should maintain the consistency between each character's unique traits and its emotional behavior. Third, it should facilitate conveying the subtle expressions of a character's internal state by the procedural generation and seamless integration of the corresponding facial animations.

Building upon the existing research results, which will be

*Rafael Bidarra and Robert Schaap are with the Computer Graphics Group, Delft University of Technology, 2628 CD Delft, The Netherlands (*corresponding author: +31 15 278 4564; e-mail: R.Bidarra@tudelft.nl, rejschaap@gmail.com).

Kim Goossens is with the Breda University of Applied Sciences, 4800 DX Breda, The Netherlands (e-mail: kimgoossens1000@gmail.com).

briefly summarized in the next section, we developed an integrated 3-layer model for the internal state of a game character that contributes to the above objectives in a number of ways. Firstly, this rich model is able to represent both the structure and the dynamics of the internal state of the character. For this reason, in this article we will metaphorically refer to this model as *the soul* of the character, as it provides the substrate, the root and the engine for every character's manifestation of (virtual) life, including its facial expressions and animations. Interestingly, the term *animation* etymologically derives from the term *anima* (Latin for *soul*), which very much conveys the same connotation we are using here. The 'anatomy' of the soul will be discussed in Section II.

Secondly, we have 'designed this soul' to be *sensitive* and *dynamic*. This means that its state can change in time when acted upon by the surroundings of the character, e.g. other characters, objects or events in its environment. For this, we had to define 'how the soul connects and interrelates' with its outside, as well as to find out how those 'soul interactions evolve' as time passes by. Naturally, this same feature also makes that the outcome of a character's emotional behavior is likely to have impact on some other character's soul (think, for example, of panic propagation). We discuss this 'I/O interface' of the soul in Section III.

Finally, we have also allowed this soul to exteriorize its internal state through the character's features. For this, we 'interfaced the soul' with a powerful procedural system for the generation and blending of facial expressions. In this way, just like with real people, subtleties in the animation of its features (e.g. eyebrows slightly quivering or frowning) are driven from the 'character's inside', its face reflecting the soul's state as on a mirror. This expressive power will be discussed in Section IV.

In the next three sections, we deal with each of these three aspects of our approach. In addition, together with describing their interplay, we also present some details of the prototype system implementing it (Section V), and illustrate them using several examples and output taken from the scenarios we have run for system evaluation (Section VI).

II. Dissecting the Soul

Research work on modeling emotions in embodied conversational agents has achieved important results in the last decade, as those by Pelachaud *et al.*, see e.g. [1][2]. More recently, Magnenat-Thalmann *et al.* have proposed models with memory to enhance such conversational agents [3]. However, their results were neither focused at nor suitable for driving game characters in real-time. We have previously discussed the integration of emotion aspects into video game characters in order to provide game players with a more vivid and compelling experience [4].

Based on observation, it is common to consider emotional behavior as mainly rooted on three major basic elements: emotion, mood and personality. Similarly, we propose to organize our interior model of a game character around three layers, based on each of those elements [4]. We will refer to this structure as the 'three layers of its soul'. Emotions are usually very short-lived. Moods tend to last longer, from several minutes to several hours. Personality can pretty much be considered constant throughout a life-time, except for minor changes that may occur over time; only exceptionally do large personality changes take place.

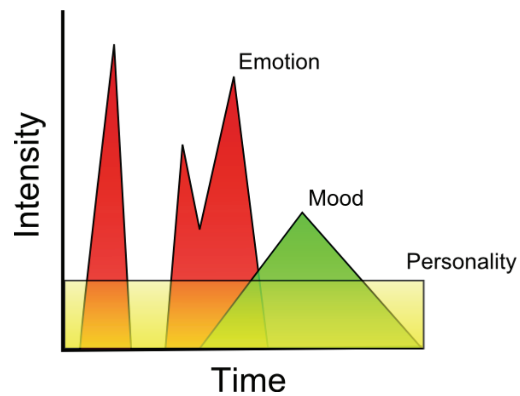


Fig. 1. How emotion, mood and personality are layered

At times, it has been discussed whether something like mood really exists, with the argument that a mood is no more than a longer lasting emotion. However, besides the difference in duration, there is a strong argument for maintaining their distinction: moods are mostly fortified or alleviated by emotions and, conversely, a given mood typically influences which emotions are invoked. Because of this interaction, it is very convenient to consider moods and emotions as separate layers of the soul. In addition, personality is an important factor of influence when it comes to determining which mood or emotion to invoke [5].

It is important to understand and describe the way in which these three layers interact. As shown in Fig. 1, the layers have an order of priority. Momentary emotions take precedence over the mood that might be present. In turn, moods take precedence over personality. Eventually, a character's behavior will be influenced first and foremost by its emotions, subsequently, by its mood and only after this by its personality. This might seem to downplay the importance of personality but, again, one must keep in mind that a character's concrete moods and emotions are strongly influenced by its particular personality [6].

A simple yet powerful model for representing emotional reactions based on three dimensions was introduced by Mehrabian: the Pleasure-Arousal-Dominance (PAD) model [7]. The PAD model is based on the view of human emotions as an input-output system, and it proposes three independent emotional dimensions or traits: Pleasure, Arousal and Dominance. In this model, the *pleasure* trait represents the amount of enjoyment or satisfaction the character is experiencing. When the character experiences something good, it will turn happy; when something bad happens, it will become sad. The *arousal* trait relates to the amount of

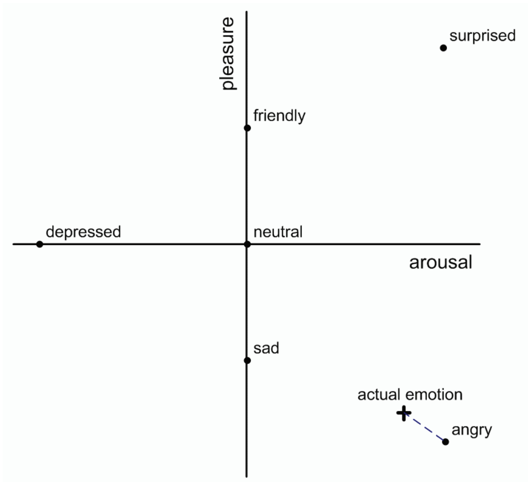


Fig. 2. Crosscut of PAD space showing the position of several emotions

excitement being experienced. For instance, when a character is receiving a lot of information, it will be aroused; when there is no information, it will be bored. Finally, the *dominance* trait relates to the feeling of being in-control and free. When a character is enslaved, its dominance will be low, when it is free to do what it pleases, its dominance will be high.

The PAD model, initially developed for convenient assessment of consumer reactions to services and products, was meant to describe and record emotions of real people. Subsequently, the model was first used in the ALMA project to track the internal state of interactive virtual agents [8]. And, just like many other researchers in this area, we also make use of the PAD model, which provides us with a very convenient foundation for keeping track of the emotions, mood and personality of video game characters.

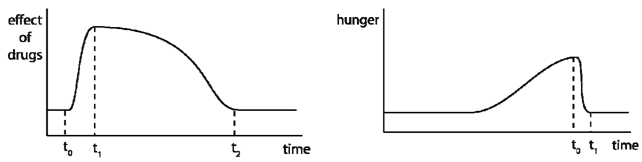
Several reasons make the PAD model very suitable for representing 'the soul of a character'. First, the PAD model is able to represent a broad range of emotions. It can be compared to creating a whole spectrum of colors using only red, green and blue. Second, the model has a low complexity due to the fact that, for representing emotions, it uses only three axes, which furthermore are almost orthogonal to each other, as we are used to, for example, in 3D space. This gives us great flexibility as it allows us to modify the values along the three scales independently without trouble. Fig. 2 shows a variety of emotions located within a crosscut of the PAD space (for simplicity, we have left out the dominance axis). Finally, in our case the PAD model also provides us with a very adequate basis for controlling the generation of subtle facial expressions.

Basing our representation of the soul of a character upon the PAD model, requires us to establish how it relates to each 'layer of its soul': emotions, mood and personality. Fortunately, valuable previous research results were very helpful in this regard:

- **emotions**: the ALMA project [8] provides us with a mapping between regions inside the PAD space and each of the 22 emotions that are defined by the OCC model [9]. In particular, this mapping allows us to determine which emotion to display, as well as its intensity.
- **mood**: PAD values are, themselves, already a suitable representation of mood. In addition, also inspired by the ALMA project, we also simulate the dynamics of moods by using a push and pull mood change function. We will illustrate its working below, but basically this function changes the current mood based on the active emotion. When the intensity of the active emotion is higher than the intensity of the current mood the current mood gets pulled towards the active emotion. When the intensity of the active emotion is lower than the intensity of the current mood the current mood gets pushed away from the active emotion, resulting in the amplification of the current mood. In this process, the intensity of the active emotion is used to determine how hard to push or pull the current mood.
- **personality**: like several other researchers, we use the Big Five, or OCEAN, one of the most popular models for representing human personality [10]. This model has had a long development process, in which many observable and permanent personality traits have been identified. Eventually, the following five traits have been selected as the most relevant: extroversion, agreeableness, conscientiousness, neuroticism and openness to new experiences. On should take care, though, as these five dimensions aren't fully orthogonal to each other. For example, there is a negative correlation between neuroticism and extroversion, resulting in that people who experience negative emotions more easily tend to be less extrovert. Merhabian himself, among others, has described the mapping we are using between PAD and the Big Five [11]. By using the Big Five model, we easily enable artists to create unique and interesting characters. Most artists are already familiar with the traits of the model, and their limited number makes it easy and intuitive to use.

III. SOUL CONNECTIONS

Human beings are strongly influenced by their environment. Sometimes we are aware of these influences, e.g. when they become more extreme, but most of the time simply we don't realize that at all. For instance, extreme heat will likely be noticed, but a mild temperature typically won't. Current computer games strive to provide richer gaming experiences. For this purpose they sometimes offer, for example, dynamic weather systems, day/night cycles or other features which strongly determine the desired ambiance for the gameplay. However, most of the time the characters in these games are blissfully unaware of what is happening around them. This makes them seem disconnected from their environment and renders their behavior unrealistic. In order to overcome this undesirable shortcoming, we came up with



(a) (b)

Fig. 3. Intensity variation over time of the effects of (a) drugs and (b) hunger on a character's internal state

a way of 'interfacing the soul' with its surroundings, taking into account the effect of various external influences when determining the internal state of a character.

A first way of achieving this consists of functionally describing, once and for all, how the effects of each such external factor evolve in time. We do this by describing how (the PAD values characterizing) the internal state of a character vary over time under those influences. For example, the effect of taking drugs will suddenly rise over a short period of time after which it will steadily fall off, returning the character back to a neutral state, as displayed in Fig. 3.a. Hunger, on the contrary, will typically build up as long as the character does not eat. Eventually, it will result in either starvation, if the elapsed period is too long, or in a quick hunger decay, as soon as the character eats some food, as illustrated in Fig. 3.b.

From the charts in Fig. 3 we can also see that this scheme flexibly allows for even more fine-grained control over the character's reactive behavior, by means of specific parameterizations. For instance, it is easy to parameterize the time it takes for a particular character to get hungry, or its wear off period for a certain drug. As often, the possibilities offered by such an approach are boundless; however, it is crucial not to create so many parameterizations that a character's behavior becomes awkward, and its control entangled in a jungle of manually set parameters.

One way of simplifying this is to explicitly make such time functions also dependent on meaningful character parameters (e.g. weight or health) and/or environment factors (e.g. lighting or narrowness). In other words, the time variation of each character's PAD values is also determined by its own internal constitution (e.g. personality) and current internal state (e.g. mood). This has the advantage that each character will automatically exhibit unique reactions to each of those influences: one character fiercely reacts to drugs, the other is more resistant to hunger, another is insensitive to darkness, or rather claustrophobe, etc.

For this research we classified a variety of such influences, and we procedurally implemented in our prototype system many of them that directly affect the mood and emotional state of a game character, including (i) common physiological aspects (e.g. fatigue, illness and hunger), (ii) the effect of (chemical) substances the character has been exposed to (e.g. alcohol, medicines and drugs), and (iii) psychological impact of various environment factors (e.g. freedom, hostility and familiarity). For each of those actual influences, we described the specific effects it has on the internal state of the character. For example, fatigue will have a negative effect on the levels of pleasure and arousal; alcohol will have a positive effect on pleasure and dominance, but a negative effect on arousal. Many new mood-influencing factors can be defined as long as their effects can be expressed in terms of changes through the PAD space.

IV. Mirror of the Soul

Facial expressions, and particularly the gaze, are very well known for conveying much of the state-of-mind of a person, and because we are all so much used to rely on them, we miss them all the more in most game characters. Most forms of expression depend on what is going on inside of the character. And because an essential factor in conveying emotion is the expression on a character's face, we use the PAD model to determine which emotions will be displayed on its face.

For a large set of *standard emotions*, their localization in PAD space is known, as for example those in the crosscut of Fig. 2. For each of those standard emotions we have defined facial expressions. To determine which facial expression should be displayed for the actual emotion (marked with a plus-sign in Fig. 2), we take the emotions that are closest to it and blend them together, using its distance to the standard emotions as a weight. For example, when the actual emotion is not too far from the anger emotion, the character will show a hint of anger; if it shifts towards the anger emotion, the intensity of the anger expression on the face will increase.

The usual approach in facial animation is to blend together static facial expressions; instead, in our approach we actually blend together facial animations. Facial animations for standard emotions can be, for example, manually created by an artist, or recorded using motion capture, or obtained in whatever other way. In any case, as long as we have a good library of the standard facial animations, we will procedurally blend them together to create more complex, vivid and appealing animations. As a part of the procedure, we typically also add some noise to the animations in order to significantly improve the character expression, because it is very unnatural for a face of a person to be completely still, even in rest. In practice, the arousal scale of the PAD model is perfectly suitable for determining how much noise to add to the facial animation: e.g. higher values of arousal, indicating that the character is very excited about something, will result in more and larger movements in the face.

The eyes have often been called 'the mirror of the soul', and that is not without a reason: they are crucial to transmit the richness of life or, in the case of game characters, to create the illusion of life. Eyes can be fixated on a single target, indicating concentration or interest; they can be moving from target to target, indicating restlessness or anxiety; or they can be staring and looking at nothing, indicating boredom.

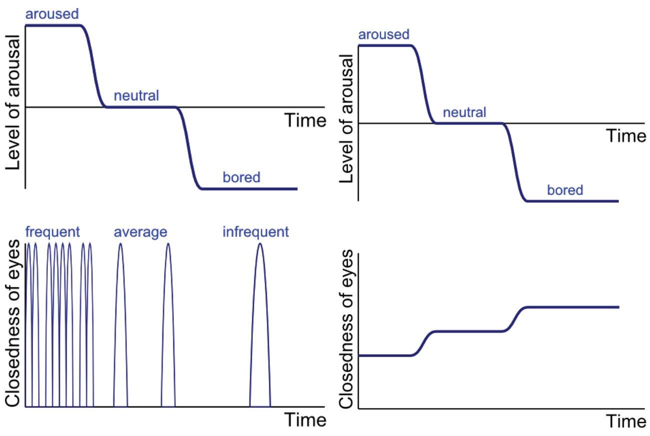
In our research, we particularly focused on the

Fig. 4. Relation between openness of eyes, gaze orientation and iris position for neutral (top), anger (middle) and surprised (bottom) emotions

expressiveness of the eyes. When a character is intensely looking at some items, or when it is specifically avoiding them, that says a lot about how the character feels about those items. One of the ways in which we influence the gaze using our internal state model is by directly relating the arousal scale to the amount of movement of the eyes. A character that is very aroused, wants to take in as much information as possible, and will be looking at many things, frequently switching focus. In addition, we also take into account the estimated amount of interest the character has in surrounding objects. So it will more frequently, and for longer periods, watch objects with a higher interest value. Conversely, when a character is very non-aroused, it will not look at objects that much, but rather stare into empty space, or look at its feet or at the floor, seldom switching focus.

Blinking plays an important role in any realistic gaze. As we will see, its frequency, speed, and regularity are also very much connected to a character's arousal. Furthermore, a character will also blink when the eyes move from one target to another or when it initiates a head movement. At the start of such movements a character will usually briefly close its eyes and, for larger movements, the character will be more likely to blink [12]. By virtue of the internal state model, our approach is able to take all these aspects into account, making it possible for characters to look much more lively.

When facing another character, the gaze of a character will be directed to certain points of interest which we called focus points (e.g. the eyes and the mouth). In our procedural implementation of the gaze controller, we used these focus points to direct the gaze during conversations between characters. In the general case, the character will typically look at the eyes and mouth, successively selected at random.



(a) (b)

Fig. 5. Development of eye blinking (a) and of eyelid closedness (b) with decreasing arousal

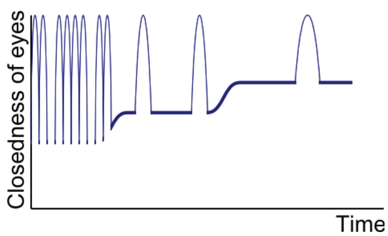


Fig. 6. Combined development of blinking and eyelid closedness

In addition, in order to provide control over the frequency and length of gaze, focus points have weights, which can be a function of the current internal state of the character.

At times, eyes and mouth can be given a higher weight value, so that they will be of higher interest and therefore more often watched. For instance, when our character is feeling very dominant when looking at the girl, the weight of the focus points on the girl's eyes will be raised, so that it more likely will look at the girl directly in the eyes. Conversely, when our character is feeling submissive, we will lower the weights of her eyes, while raising the weight of another focus point below her face. This will suggest that our character is not looking straight at her, but slightly down in a submissive way. In turn, when our character is pensive, the weight of a focus point above her head is increased, as people have the tendency to look up while thinking. Finally, interpersonal gaze was also made dependent on personality: an introverted character, for instance, is more likely to avoid looking at the eyes directly than an extroverted character. Therefore, a character's personality has also some effect on the initial weights given to focus points.

Eyelid movements are controlled by several aspects of the character's internal state. First, characters should blink every once in a while, and the blinking frequency is influenced by the level of arousal the character is experiencing. Arousal is also determined by the openness of the eyelids: a highly aroused character will have its eyes more open than one who is drowsy or bored. In the latter case, the eyelids will be droopy and more closed. Finally, the direction the character is looking at has also influence on the openness of the eyelids; for example, when the character is looking up, the eyelids will be more open than when the character is looking down. As a rule of thumb, in a neutral state the eyelids cover a small fraction of the iris, and this position is maintained as the eye changes direction, making the eyes look more natural. Finally, the current level of arousal also affects the fraction of the eye that is covered by an eyelid. Fig. 4 illustrates how the eyelids keep a small part of the iris covered for different emotions. When the girl is surprised (below), her level of arousal will be very high, resulting in her eyelids being wide open. Still, when she is looking up or down while surprised, the eyelids will slightly open or close to maintain the same relative position to the iris.

To further improve character liveliness, blinking is also controlled by the arousal scale in both its speed, its frequency and the amount of noise. A very aroused character will open its eyes faster and blink more often, as illustrated in Fig. 5.a. In contrast, a non-aroused character will blink less often, the movements of the eyelids will be slower and the eyes will be closed for a longer period of time, making it look bored or sleepy. Fig. 5.b illustrates the relationship between the level of arousal and the closedness of the eyes. When the character is very aroused, the eyes will be more open, whereas when the character is very bored the eyes will be more closed. The temporal result of combining both effects is depicted in Fig. 6.

V. PROTOTYPE SYSTEM ARCHITECTURE

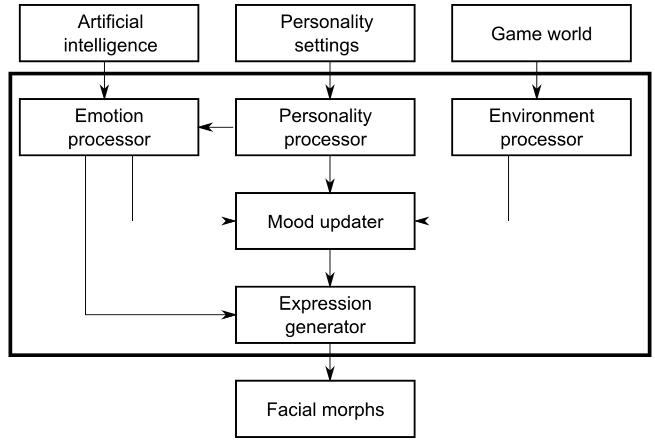


Fig. 7. Architecture of our prototype system

Fig. 7 gives an overview of the architecture of our prototype system; see also [13] for more technical details. The thick black line denotes the boundary of the system. Blocks outside it denote other game modules, including e.g. artificial intelligence, the personality settings and the state of the game world, which provide the system with the necessary information to maintain the internal state of video game characters. The module facial morphs, at the bottom, is responsible for displaying facial expressions.

Inside the system, the first component is the *emotion processor*, which is responsible for evaluating the input received from the artificial intelligence module. It determines

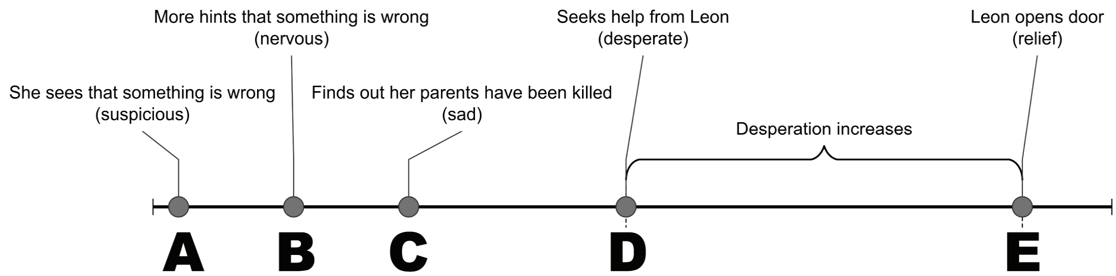


Fig. 8. Timeline of our reenacted movie scene (clip available online at http://graphics.tudelft.nl/~rafa/video.cig10.mp4)

which emotion should be invoked, and computes the intensity this should have. For this, it needs to take into account the personality of the game character.

The second component in our system is the *personality processor*, which takes as input the personality settings, externally specified by the character designer using the Big Five model. The personality processor converts this model to PAD values, which are then internally used in the system.

The *environment processor* component takes into account the influence of all relevant factors supplied by the game state and reduces them to usable PAD values for our system. For example, imagine some construction workers are making a lot of noise on the street, but our character is at home. As long as the character is inside with closed windows, the noise has little to no effect on it; however, by opening a window or a door, all that noise enters the room and the character will experience displeasure and anxiety.

The *mood updater* component is responsible for integrating all the above information, in order to maintain a consistent mood for the video game character.

Finally, the *expression generator* component is responsible for driving the facial morphs, which we have based on the FACS model [14]. Using FACS facial building blocks, we have built a library of facial animations, which are then blended together as a function of the internal state of the character.

To implement our prototype we chose Houdini, a 3D modeling and animation package by Side Effects Software Inc. Traditionally, the strength of Houdini lays in its visual effects capabilities, which is why it has been used in Hollywood blockbusters as, for example, the latest Harry Potter or Spiderman movies.

One of the reasons Houdini has been so popular and powerful for visual effects, modeling and animation is because of its unique procedural approach. It allows artists to rapidly prototype their ideas, and refine them from there until they are satisfied. When your project has grown very complex, and you are not happy about some of your first steps, it is easy to go back and make the appropriate changes. The procedural nature of Houdini makes sure that these changes get propagated throughout your project. For example, in the beginning of our implementation, we used a very basic and crude face, using some low quality facial animations for testing, which allowed us to quickly get started. Later on, when we managed to produce some high quality animations, it was just a matter of plugging them in and re-run our sample script.

The system has been implemented with a minimal amount of scripting and programming due to the node-based workflow Houdini provides, enabling us to incrementally create, wire and refine networks of nodes to implement the desired functionality. Working with Houdini allowed us to be very productive, very quickly, so we definitely recommend using it in similar projects in this research area. The resulting animations run practically in real-time on a high-end PC, which is quite impressive considering the fact that the whole system was declaratively implemented.

VI. Evaluation

To evaluate our model we used our prototype system to reenact some intense scenes from classic movies. One of these scenes was taken from the movie “Léon” directed by Luc Besson, featuring Jean Reno, Gary Oldman and Natalie Portman. In Fig. 8 you can see the timeline of events occurring in this particular reenacted scene (also available online at http://graphics.tudelft.nl/~rafa/video.cig10.mp4). The movie is about Mathilda, a 12-year old girl living in New York. The scene starts with Mathilda coming home from doing some groceries for her parents. At first she sees some shattered glass on the floor and starts suspecting something is wrong (A). As she continues to her family's apartment (B), she finds out that her family has been brutally murdered (C). The gangsters are waiting outside of the apartment. She pretends that she does not live there and walks past them to her neighbor's apartment (D). Her neighbor is Léon, she knocks on his door but there is no response. She starts to get nervous because the gangsters are still looking at her. Her desperation increases as time goes by. When Léon at last opens the door to let her inside, she feels extremely relieved (E).

We have used this movie scene to evaluate the influence of noise, mood, personality and environment separately. For this, we showed a number of participants a few short movies, displaying two versions of the same scene side by side; one of them had a certain parameter altered or turned off corresponding to the technique under evaluation. After watching each movie, the participants were asked to answer some questions about it. Here we will just briefly describe three of these evaluation tests and their main outcomes.

In the first movie, to evaluate how noise influences our perception of emotions, one of the scene versions had no noise added. Most people did not really feel that either scene was more realistic than the other, but they did find that Mathilda without noise was slightly more robotic than Mathilda with noise. They also seemed to agree that Mathilda with added noise had more natural eye blinking than Mathilda without noise, thus confirming that a too regular blinking makes a character look robotic. Finally, people strongly agreed that the movements of the mouth for Mathilda with noise are more realistic than for Mathilda without noise.

In the second movie we evaluated the influence of the mood updater, which we turned off in one of the scene versions. Most people found that the scene with the mood updater turned on was slightly more compelling than the one with it turned off, stating that the emotions of Mathilda with the mood updater are quite consistent and recognizable. People also said that none of the characters was acting very dramatically, thus indicating that the mood updater has little effect on that feature. But when asked whether the expression changes were smooth, people tend to agree that they are a lot smoother with the mood updater turned on. This makes sense because the mood updater just tends to make changes in emotion less sudden: when you are in a sad mood, it will take something very positive to happen before you display a big smile; otherwise, it is more likely that you will simply put on a faint smile and then fall back to your original sad expression, unless the positive event actually manages to pull enough your mood.

In the third movie, Mathilda's personality was varied: in one of the versions she was very much introvert, and in the other, extrovert. People felt very strongly that the extroverted Mathilda was more interesting to look at, and also that she was more emotional than the introverted character. It seems that the effect of this personality trait is very effective, as this is congruent with how people tend to think that extroverted people are more appealing, and definitely more emotional, in the sense that they show their inner feelings more than introverted people. People also felt very strongly that the extroverted Mathilda was more affected by what was happening. Interestingly, when explicitly asked whether the introverted Mathilda was more introverted, people also confirmed that very strongly. Finally, when asked which Mathilda was acting more realistically, people seemed to agree that the introverted Mathilda was acting slightly more realistically. However, it is hard to conclude much about this because the participants did not actually see what Mathilda was reacting to.

VII. Conclusion

In conclusion, the approach we have described in this article proposes to enrich game characters' expressiveness by combining procedural facial expression generation with a rich internal state model, which capitalizes on previous results and models used e.g. for conversational agents. This methodology presents several advantages. First, it allows game designers and artists to also 'design the soul' of a game character, using intuitive parameters to declare how its unique 'interior world' is expected to express itself. Second, this model of a character's interior state has been interfaced with the outside world, enabling it to interact with a variety of external factors, objects and circumstances. This results in emotional behaviors developing naturally over time due to a variety of influences, resembling the dynamics of life itself. A procedural mood updater assures that the character’s unique traits are kept consistent throughout all interactions. Third, this approach grounds the procedural generation and blending of a character's facial animations in the 'richness of its soul', allowing for a very vivid and compelling portrayal of a large variety of subtleties, including complex and even somewhat ambiguous expressions.

References

[1] C. Pelachaud and I. Poggi, “Subtleties of Facial Expressions in Embodied Agents,” *Journal of Visualization and Computer Animation*, vol. 13, pp. 301-312, 2002.

[2] F. de Rosis, C. Pelachaud, I. Poggi, V. Carofiglio and N. De Carolis, “From Greta’s Mind to her Face: Modeling the Dynamics of Affective States in a Conversational Embodied Agent,” The International Journal of Human-Computer Studies, vol. 59 (1-2), July 2003

[3] Z. Kasap, M. Benmoussa, P. Chaudhuri and N. Magnenat-Thalmann, “Making Them Remember – Emotional Virtual Characters with Memory”. *IEEE Computer Graphics and Applications*, *29(2)*, pp. 20-29, March 2009.

[4] R. Schaap and R. Bidarra, “Towards Emotional Characters in Computer Games,” in *Proceedings of the 7th International Conference on Entertainment Computing* (Pittsburgh, PA, September 25 - 27, 2008). S. M. Stevens and S. J. Saldamarco, Eds. Lecture Notes in Computer Science, vol. 5309. Springer-Verlag, Berlin, Heidelberg, pp. 167–172. DOI=http://dx.doi.org/10.1007/978-3-540-89222-9_19

[5] C. Becker, S. Kopp, and I. Wachsmuth, “Simulating the emotion dynamics of a multimodal conversational agent,” in *ADS*, pp. 154–165, 2004.

[6] K. Perlin, “Real Time Responsive Animation with Personality”, *IEEE Transactions on Visualization and Computer Graphics*, 1(1), 1995.

[7] A. Mehrabian, “The three dimensions of emotional reaction,” Psychology Today, 57–61 (1976)

[8] P. Gebhard, “ALMA: A Layered Model of Affect,” in *AAMAS 2005: Proc. of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems,* ACM Press, New York, 2005, pp. 29–36.

[9] A. Ortony, G. Clore, and A. Collins,. “The Cognitive Structure of Emotions,” Cambridge: Cambridge University Press, 1988.

[10] O. P. John and S. Srivastava, “The big-five trait taxonomy: History, measurement, and theoretical perspectives,” in *Handbook of Personality: Theory and Research*, New York, USA, 1999. Guilford.

[11] A. Mehrabian, “Analysis of affiliation-related traits in terms of the pad temperament model,” Journal of Psychology, 1997, pp. 101–117.

[12] C. Peters and C. O’Sullivan, “Attention-driven eye gaze and blinking for virtual humans”, in ACM SIGGRAPH 2003 Sketches and Applications, New York, NY, USA, 2003. ACM Press, , pages 1–1.

[13] R. Schaap, “Enhancing video game characters with emotion, mood and personality,” MSc Thesis, Delft University of Technology, Delft, The Netherlands, February 2009.

[14] P. Ekman, “Basic emotions,” in *The Handbook of Cognition and Emotion*, John Wiley & Sons, Ltd., 1999, pp. 45–60.