Learning Emotions in Virtual Environments

Mannes Poel, Rieks op den Akker, Anton Nijholt and Aard-Jan van Kesteren
Department of Computer Science, University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
email: {mpoel,infrieks,anijholt}@cs.utwente.nl

Abstract

A modular hybrid neural network architecture, called SHAME, for emotion learning is introduced. The system learns from annotated data how the emotional state is generated and changes due to internal and external stimuli. Part of the modular architecture is domain independent and part must be adapted to the domain under consideration. The generation and learning of emotions is based on the event appraisal model.
The architecture is implemented in a prototype consisting of agents trying to survive in a virtual world. An evaluation of this prototype shows that the architecture is capable of generating natural emotions and furthermore that training of the neural network modules in the architecture is computationally feasible.
Keywords: hybrid neural systems, emotions, learning, agents.

1 Introduction

We have developed a virtual reality environment [Nijholt and Hulstijn, 2000] that we are using as a laboratory to develop and implement ideas about human-agent and agent-agent interaction in such visualized environments. This environment is meant to develop in a virtual interest community where people can represent themselves and can explore and interact, not only with each other, but also with community agents with task and domain knowledge. Our aim is to build an inhabited world where is it difficult, if not impossible, to distinguish between agents that are somehow (semi-)controlled by humans and agents that act autonomously (in interaction with the environment and its other inhabitants). Obviously, research results can be used as well in a virtual environment where there is one-to-one communication between an (embodied) interface agent and a user. Several agents have already been introduced into this environment. A Java based agent framework has been introduced to provide the protocol for communication between agents, also allowing the introduction of other agents. In fact, we can have a multitude of useful agents, where some just trigger an animation, some can walk around and some have built-in intelligence that allows them to execute certain actions based on interactions with visitors.

An agent is called believable, if some version of a personality shows in the interaction with a human. Main requirements for believability are: personality, emotion, self motivation, social relationships and consistency of expression. Embodiment makes it possible to show facial expressions, body language, lip movements and gestures that support interaction. It allows also, more than "just" language, the expression of emotional behavior in which personality shows. Initially we do not exclude any human activity or task from the embodied agents in our (future) environments. An agent may solve a problem individually, it may negotiate with others to solve a problem, it may be involved in a creativity demanding task, it may feel and express sympathy for others or just fall in love with a visitor or another agent. Clearly, the latter "tasks" ask for some modelling of emotions, but that is also the case for the former.

The more obvious physical aspects of emotion that can be generated in embodied agents and that can be interpreted by humans and other embodied agents can, in accordance with most authors, for example Picard [Picard, 1997], be distinguished in a list of easily perceivable bodily components of emotions (facial expressions, voice intonation, gestures and movements, posture, and pupilary dilation) and a list of components less apparent to others (respiration, heart rate, pulse, temperature, perspiration, muscle action potentials and blood pressure). Although input devices become available to measure values from the second list and feed them into the computer (or even, using haptic and tactile devices, to have embodied agents output and display some of these components).

Most of our research in the area of embodied agents deals with intelligence [Egges *et al.*, 2001] and facial expressions [Duy *et al.*, 2001]. In this paper we introduce a model that makes it possible to talk about an emotional state and emotional state changes because of appraisals of events that the agents perceive in their environment. It is based on the the OCC model [Ortony *et al.*, 1988], a cognitive theory for calculating cognitive aspects of emotions. In this paper we use the OCC model as the basis for the supervised learning approach of emotions we advocate. The OCC model has also been used, in a very stripped-down version, in the Oz project [Reilly and Bates, 1992] that was concerned with the development of a theater world inhabited by emotional agents.

Another use of the OCC model was made in the Affective Reasoner of Elliot [Elliot, 1993; 1994]. The Affective Reasoner maps an event to emotion types by using a rule based system. Our system maps an event to a vector of emotion intensities, using a hybrid neural network architecture.

In order to design our model and to experiment with it we had to design a much more simple environment than the environment which we discussed above.

The simplified environment and the simplified agents will be introduced below. Events that are appraised in this simplified environment are a long way from observing or deriving an emotion from generated speech, from a facial expression or from a bodily posture. Rather we have agents that look for water, apples and health and that have to deal with predators. Nevertheless, we assume that in the future it will be possible to use the model in our virtual environments inhabited by interacting embodied agents as well.

In section 2 an overview of the architecture for emotion learning is given. Then the prototype is discussed in section 3. Afterwards some test results are reported in section 4 and finally the conclusions are given in section 5.

2 The architecture for emotion learning

In this section we will give a global overview of the hybrid adaptive architecture for learning emotion generation. The hybrid architecture has certain neural network components which must be trained using annotated data. Obtaining annotated data from the interaction of agents in a virtual world is tedious and time consuming. Therefore we opted for a hybrid architecture, called SHAME (Scalable, Hybrid Architecture for the Mimicry of Emotions) that has the potential to keep the amount of necessary training data small. The most important aspects of the architecture will be discussed in the subsequent subsections. More information on the SHAME architecture can be found in the work of van Kesteren and Poel [Kesteren, 2001; Poel and Kesteren, 2001].

2.1 Global overview of the architecture

Our architecture is based in the OCC model [Ortony *et al.*, 1988] of emotions. This is an event appraisal model, meaning that changes to the emotional state are event driven. According to the OCC model every emotion intensity can be represented by a scalar. We deviate slightly from this view by representing each pair of positive and negative emotions, such as for example joy and distress, by one scalar, the intensity value. A positive (negative) intensity value corresponds to the positive (negative) emotion. An overview of the different emotions can be found in table 1.

Pos. emotion types	**Neg. emotion types**
Joy	Distress
Hope	Fear
Satisfaction	Fears-confirmed
Relief	Disappointment
Pride	Shame
Admiration	Reproach
Happy-for	Resentment
Gloating	Pity
Love	Hate

Table 1: The positive (Pos.) and negative (Neg.) emotion types according to the OCC model

Our hybrid architecture SHAME should implement the OCC model, so from a functional point of view it should map an event and an old emotional state to a new emotional state. Of course this mapping depends on the properties of the event under consideration. The global overview of the SHAME architecture is depicted in figure 1.

The emotional state is described by a vector $\langle e_1, e_2, \ldots e_n \rangle$ of length $n = 9$ where each component e_i refers to an emotion pair described in table 1. If $e_i > 0$ ($e_i < 0$) then the positive (negative) emotion is experienced with intensity e_i ($-e_i$). Due to this design choice the model cannot generate positive and negative emotions of the same pair, for instance joy and distress, simultaneously. Furthermore we assume that $-100 \leq e_i \leq 100$. The (normalized) emotion impulse vector is a vector describing the impulse, force, on the emotional state, This emotion impulse is used by the emotional state calculator to calculate the new emotional state and is of the form $\langle ei_1, ei_2, \ldots ei_n \rangle$ where ei_i is the emotion impulse for the emotion pair corresponding to e_i. Again we assume $-100 < ei_i < 100$.

In the first phase the emotional meaning of an event is appraised, i.e. determining the emotional effect (impulse) of the event on the emotional state. This is done by determining the type of the event. In the prototype of section 3 examples of event types are: new apple spotted, other agent took apple, I am going to flee, etc. Based on this event type an event appraiser is selected. Each (relevant) event type has exactly one corresponding event appraiser. Then, secondly, the event, together with relevant event information is sent to the event appraiser. This event appraiser calculates, based on the event information, the emotional impulse of the event. This is stored in the emotional impulse vector (EIV). For example in our prototype, agents have to survive and therefore they need, among other things, to eat. In the case the agent takes an apple, which is related to the well-being of the agent, then according to the OCC model the only non zero emotion impulse is joy/distress and the emotion impulse is given by $100 - \#food_{self}$, where $\#food_{self}$ is the amount of food the agent has (and is limited to 100). In particular the joy impulse vector is nega-

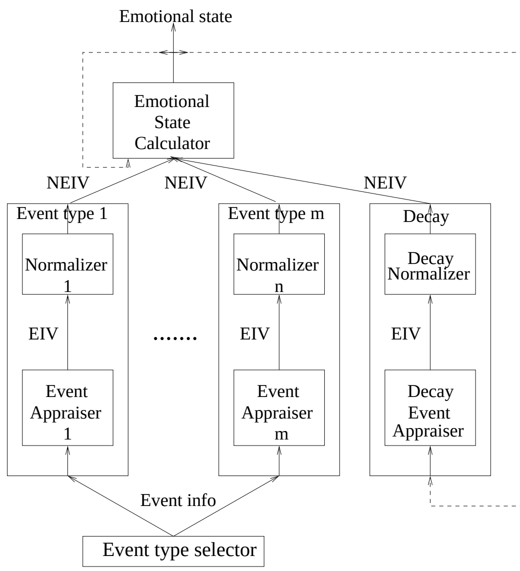

Figure 1: The SHAME architecture, the Event Appraisers calculate an emotion impulse vector EIV which is normalized to a normalized emotion impulse vector $NEIV$ by the Normalizers a dashed line represents one unit time delay.

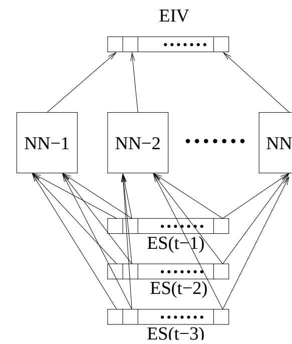

Figure 2: The architecture for the decay event appraiser.

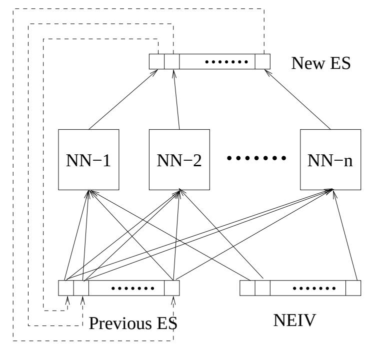

Figure 3: The architecture for the emotional state calculator (ESC).

tive correlated with the amount of food and since the amount of food is always less than 100 it is always a joy impulse. Finally the new emotional state is calculated by the Emotional State Calculator (ESC). This new emotional state depends on the old emotional state and the normalized emotion impulse vector.

The event appraisers appraise the emotional impulse of the event in a qualitative way and the normalizers form an interface between the ESC and the event appraisers, in such a way that, firstly, the ESC can treat all event appraisers in a uniform way and, secondly, it makes the *ESC domain independent*. Observe that the event appraisers and normalizers are domain specific.

A special event is the decay event, which models the decay of emotions over time, i.e. temporal phenomena. These events occur after a fixed amount of time and depend on the previous emotional states. The neural network architecture for the decay event is given in figure 2. Observe that there is a separate feed-forward neural network for each component of the EIV. The inputs of each of the neural networks and the number of needed previous emotional states must be determined by hand, using a-priori knowledge. Therefore the neural networks can be kept small and less training data is needed.

2.2 The neural network architecture for the *ESC*

The neural network architecture for the ESC is depicted at the right in figure 3. The functionality of the ESC is to calculate the new emotional state (New ES) given the old emotional state (Previous ES) and the normalized emotional impulse vector ($NEIV$). In order to keep the networks relatively small we opted again for a separate neural network for each component of the emotional state. Of course we could take an Elman network [Elman, 1990], which can learn temporal phenomena. But then again we will need a large network and hence a lot of training data, also due to feedback loops in the Elman network.

This concludes our overview of the general architecture.

3 The prototype

In order to develop and test a system that learns to generate emotions, an environment is needed, which is inhabited by agents that can show emotions. As we make use of the OCC model [Ortony *et al.*, 1988] as basis for our AI-model of emotion, it is important that the agents have an explicit or implicit representation of goals, standards and attitudes. Agents should be able to translate observations into terms of these three concepts.

3.1 The virtual world

The domain is a grid-world containing grass, water pools that can be dry or contain water, apple trees that can bear apples and rocks, possibly with herbs growing on them. The status of the trees, water pools and rocks constantly changes in a non-deterministic way. Multiple agents and predators inhabit this virtual world. An agent can only see a small part of the world and only knows where the visible agents and predators are. He knows the location of all the trees, rocks and water pools, but the status is only known for the visible trees, rocks and water pools. An important property of the environment is, that it contains multiple agents, as some emotions depend on actions of other agents or on the consequences of events for other agents. As the user has to annotate natural emotional behavior, it's important that the user is able to put himself in the position of an agent. A user can only do this, if the behavior of the agent is believable enough and if the user has the exact same knowledge as the agent.

3.2 The agents

The goal of the agents is to survive, therefore they need food and water, which can be supplied by apple trees and water pools. There are predators that can be dangerous for the agents. An attack by a predator affects the health of an agent. An agent can regain health by eating a herb. Moreover agents have the capability to appraise events. For instance, they can see whether an event satisfies a particular goal or has a positive or negative effect on the probability that a particular goal will be satisfied.

The agents have expectations, which are important for emotion types such as hope and fear. They have some kind of memory about previous expectations and are able to compare new events to these previous expectations (important for the emotion types relief, fears-confirmed, satisfaction and disappointment). The same kind of reasoning the agents are able to apply for themselves, they can also apply for other agents (important for emotion types happy-for, resentment, gloating and pity). Agents can compare their own actions or those of other agents to their standards (important for emotion types pride, shame, admiration and reproach).

To a large extent, the behavior of a predator is random. The only exception is when a predator is in the neighborhood of food, water or a herb. Then it will stay there because it knows that agents will come there sooner or later. This means that the presence of a predator is an indication that food, water or a herb may be nearby. On the other hand the presence of food, water or a herb is an indication that a predator may be nearby. There is a social order between agents and they can choose (dependent on their character) either to follow the leader of a group or to go their own way. Social grouping is a result of common concerns. An agent knows how thirsty, hungry and healthy the agents in his group are.

The character of the agent, determines the action it will undertake in response to the events that occur. A character is defined by four character traits: bravery, generosity, sociability and compliance. In this virtual world there are two agents: Hero and Grumph. A personality defines what kind of emotions they can experience and how strong. Hero is self-confident and idealistic, he feels emotions strongly, in particular positive emotions will last a long time. He thinks that killing a predator is praiseworthy. Grumph is introvert and does not enjoy life; disappointment and negative social events have a strong impact on his emotional state. He thinks that other agents should think about him and not about themselves. Killing a predator is only praiseworthy if he himself does it. Moreover both agents have the following standards:

- The agent with the least food (health) should get the next apple (herb).
- The healthiest agent should attack the predator, if any.
- Feeling the emotions resentment and gloating is not praiseworthy.
- Feeling the emotions happy-for and pity is praiseworthy.

Table 2 defines the personality, more precisely the emotional characteristics, of Hero and Grumph. These personalities, emotional characteristics, are used by the trainee to annotate the training data. The goals and standards together with the personality are also used to design, by hand, the event appraisers.

Although love and hate are important emotion types, implementing these emotion types would not only have meant a more complicated ESC and more complicated event appraisers, but also more requirements to the cognitive capacities of the two agents. After all, to be able to experience love and hate, a cognitive representation of attitudes is necessary. Therefore the emotion types love and hate have been left out in order to keep the prototype somewhat simpler

	Hero	**Grumph**
Joy	+, short	−, long
Distress	+, short	−, long
Hope	+, long	−, long
Fear	+, short	−, long
Relief	+, short	−, short
Disappoint.	+, short	+, long
Pride	+, long	−, short
Shame	+, long	−, long
Admiration	+, long	u.e.
Reproach	+, long	+, long
Happy-for	−, short	−, short
Resentment	−, short	−, long
Gloating	u.e.	+, long
Pity	−, short	−, short

Table 2: Definition of the personalities. The strength and duration of the various emotions. + (−) means that the emotion will be experienced relatively strongly (weakly). U.e. means that the agent is unable to experience the emotion. Short (long) indicates a short (long) duration of the emotion

The event appraisers are constructed by hand. For example in the case that Hero or Grumph took (eat) an apple the only non-zero component of emotion emotion impulse vector is the joy/distress component, and the value is given by

$$100 - \#food_{self},$$

where $\#food_{self}$ is an internal variable (state) of the agent coding the amount of food an agent has. In the case that Hero or Grumph wants an apple to only non-zero component of the emotion impulse vector is the pride/shame component, and the impulse for Hero is given by

$$\#food_{Grumph} - \#food_{self},$$

where $\#food_{Grumph}$ is an internal variable of Hero coding the amount of food of Grumph.

More details can be found in [Kesteren, 2001]. The parameters for the neural network architecture and the amount of training data used are summarized in table 3. As we can see from table 3 the neural networks

Emotion type	nhn	nad
ESC		
joy/distress	6	250
hope/fear	10	250
relief/disapp.	6	250
pride/shame	6	250
admir./reproach	6	250
happy-for/resent.	6	250
gloating/pity	10	250
decay event appraiser		
joy/distress	5	300
hope/fear	5	250
relief/disapp.	5 (2)	100
pride/shame	5	300
admir./reproach	1 (2)	100
happy-for/resent.	2	100
gloating/pity	2	100

Table 3: Number of hidden neurons (nhn) and the amount of annotated data (nad) used. If there is a difference between Hero and Grumph then the number between brackets corresponds to the architecture of Grumph.

are kept small and only a relatively small amount of training data is used. Training is done in a standard way except for the normalizers. To train the normalizers, $(EIV, NEIV)$ pairs are needed. The EIV's can be calculated once the event appraisers are known, but the $NEIV$ can not be calculated directly from the emotional state because then one needs a complete understanding of the ESC. In this prototype an approximation of the inverse of the trained ESC is determined and used in order to generate $(EIV, NEIV)$ pairs.

4 Test results

After training the agents Hero and Grumph we tested the system with different scenarios. A scenario determines a particular situation in the world (like a group of agents spots an apple), and an indication of the attitude of each agents of different personalities towards the situation (whether they want to take the apple or not). The experiments involve a comparison of the emotional states of the personalities after the agents performed an action (an agent takes the apple) with the characteristics of the personalities shown in table 1. Among the tested scenarios are: the agent takes an apple himself, the other agents takes an apple, being chased by four predators, the agent is attacked by a predator. One of the tested scenarios is discussed in detail below.

One of the scenarios is that Hero and Grumph spot a predator, which one agent deliberately attacks and kills. The other agent flees. In the case that the agent himself is weak and the other agent has 100% health, the fact that the agent himself is going to attack leads to pride and the intention of the other agent to flee, leads to reproach, cf. figure 4. The following events

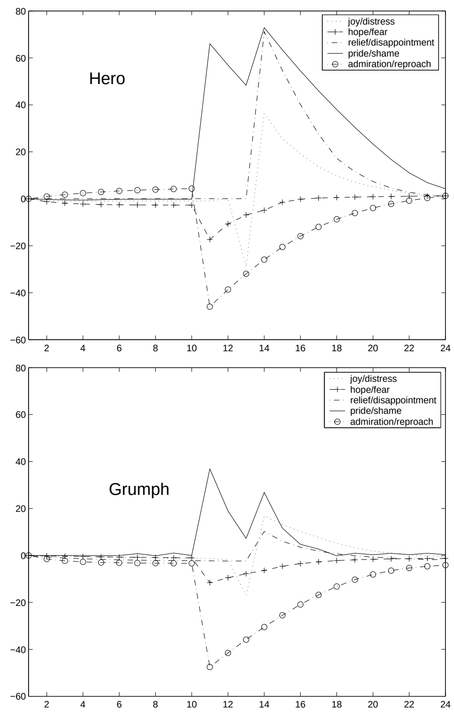

Figure 4: The most strongly experienced emotions of Hero (top) and Grumph (bottom) if the agent attacks a predator himself. The agent himself is very healthy and the other is weak.

have occurred: the first 10 turns nothing happens, the 11th turn a group of two agents, Hero and Grumph, spot a new predator. The agent himself declares that he will attack the predator and the other agent declares that he will flee from the predator. The 14th turn, the agent himself was attacked by the predator. The 15th turn, the agent himself was attacked again by the predator, he killed the predator and “the predators lost” event occurred. As can be seen, killing the predator also leads to pride. If Hero spots the predator, he experiences only a small amount of fear, as Hero does not expect to attack the predator himself. Grumph does not experience much fear either, but that is just because fear is not a strong emotion for him. Distress is experienced if the predator attacks the agent, but joy is experienced if the predator has been killed. Finally, relief is experienced if the agent observes that no more predators are nearby. In the case that the other agent is weak also, Hero expects to attack the predator himself and therefore he experiences more fear in this case. Hardly any pride is experienced as a result of the intention to attack, as there is nothing special about this intention in this case. For the same reason, no reproach is experienced. The observed emotional characteristics, decay and average intensity, over all tested scenarios are given in table 4. As can be seen from table 4 emotions specified to have a short duration, cf. table 2, have an average decay time of approximately 8 and all decay times are below 9 and emotions specified to have a long duration all have a decay time above 9 and the average decay time is approximately 16.

	Decay		**Av. int.**	
Emotion type	**H**	**G**	**H**	**G**
joy	7	16	25.49	17.8
distress	7	18	29.5	16.3
hope	11	18	20.3	14.0
fear	8	19	28.7	21.8
relief	9	8	35.5	15.8
disapp.	9	20	17.5	16.8
pride	14	7	28.0	17.1
shame	14	19	43.3	20.0
admir.	18	4	20.5	0.0
reproach	15	18	24.2	18.4
happy-for	8	7	13.1	12.5
resent.	8	9	14.3	22.8
gloating	3	21	0.0	14.0
pity	9	8	16.2	0.0

Table 4: The observed emotional characteristics, decay and average intensity (Av. int.), for Hero (H) and Grumph (G).

From the average intensities it follows that there is a clear difference between emotions that must be experienced relatively strongly and emotions that must be experienced relatively weakly. Hence the personalities in the prototype behave as specified in table 2.

5 Conclusions

In this paper an emotion theory based, hybrid architecture, called SHAME, which can learn to generate emotions was presented.

Complex emotional phenomena can be modelled using the SHAME architecture. Significant parts of the architecture are domain independent, meaning that these only have to be designed once for a particular personality and can be reused easily. The architecture is distributed, which gives the advantage that scalability with respect to event types is no problem.

A comprehensive prototype with 14 emotions has been implemented and successfully tested using the architecture. Complex emotional behavior can be observed, whereas the implementation is kept relatively simple.

Whether the prototype really functions in a natural way has not yet been demonstrated to our complete satisfaction, but is a topic for future research.

References

[Duy *et al.*, 2001] Bui the Duy, D. Heylen, M. Poel, and A. Nijholt. Generation of facial expressions from emotion using a fuzzy rule based system. In M. Stumptner, D. Corbett, and M. Brooks, editors, *Proc. 14th Australian Conference on Artificial Intelligence (AI'01), Lecture Notes in Artificial Intelligence 2256*. Springer, 2001.

[Egges *et al.*, 2001] A. Egges, A. Nijholt, and R. op den Akker. Dialogs with BDP agents in virtual environments. In *Proceedings 2nd IJCAI workshop on Knowledge and Reasoning in Practical Dialogue Systems*, 2001. To appear.

[Elliot, 1993] C. Elliot. Using the affective reasoner to support social simulations. In *Proc. of the 13th International Joint Conference on Artificial Intelligence*, pages 194–200, 1993.

[Elliot, 1994] C. Elliot. Research problems in the use of a shallow artificial intelligence model of personality and emotion. In *Proc. of the 12th National Conference on Artificial Intelligence*, pages 9–15, 1994.

[Elman, 1990] J.L. Elman. Finding structure in time. *Cognitive Science*, 14:179–211, 1990.

[Kesteren, 2001] A.-J. van Kesteren. A supervised machine-learning approach to artificial emotions. Master's thesis, Department of Computer Science, University of Twente, 2001.

[Nijholt and Hulstijn, 2000] A. Nijholt and J. Hulstijn. Multimodal interactions with agents in virtual worlds. In N. Kasabov, editor, *Future Directions for Intelligent Systems and Information Science*, Studies in Fuzziness and Soft Computing, chapter 8, pages 148–173. Physica-Verlag, 2000.

[Ortony *et al.*, 1988] A. Ortony, G.L. Clore, and A. Collins. *The Cognitive Structure of Emotions.* Cambridge University Press, 1988.

[Picard, 1997] R. Picard. *Affective Computing.* MIT Press, 1997.

[Poel and Kesteren, 2001] M. Poel and A.-J. van Kesteren. A modular hybrid architecture for emotion generation. In *Proc. conference on Artificial Neural Networks and Expert Systems, ANNES'2001.* to appear, 2001.

[Reilly and Bates, 1992] W. Reilly and J. Bates. Building emotional agents. Technical Report CMU-CS-92-143, Carnegie Mellon University, 1992.