# Emotion-based Agents: Three approaches to implementation (Preliminary Report)

**Rodrigo Ventura** and **Carlos Pinto-Ferreira**
Instituto de Sistemas e Robótica
Instituto Superior Técnico
Rua Rovisco Pais, 1
1049-001 Lisboa
Portugal
emails: {yoda,cpf}@isr.ist.utl.pt

### Abstract

This paper describes three implementations of an emotion-based agent architecture previously described in (Ventura & Pinto-Ferreira 1998a; Ventura, Custódio, & Pinto-Ferreira 1998a; Ventura & Pinto-Ferreira 1998b; Ventura, Custódio, & Pinto-Ferreira 1998b). This architecture is based upon the Damasio findings on the neurophysiological foundations of human emotions (Damasio 1994). The underlying model is briefly described in the first two sections. Then, the implementations are presented, along with some experimental results.

## Introduction

According to Damasio (Damasio 1994), some aspects of human intelligence, namely the ability to make appropriate decisions in dynamic, complex, and unpredictable environments, depend on emotions. This paper follows previous theoretical work (Ventura & Pinto-Ferreira 1998a; Ventura, Custódio, & Pinto-Ferreira 1998a; Ventura & Pinto-Ferreira 1998b; Ventura, Custódio, & Pinto-Ferreira 1998b) on developing a *prescriptive* model of emotion-based agents. This model is distinguished from the *descriptive* model presented by Damasio, which as a consequence, is much more abstract and complex, in the sense of attempting to describe the human brain.

To validate the referred theoretical framework, three implementations were constructed and experimented, using different approaches. The obtained results are presented, showing some aspects of the proposed model.

The seminal publications of Sloman (Sloman & Croucher 1981) and Minsky (Minsky 1988) stated the need to research emotions in the context of Artificial Intelligence. However, the field of artificial emotions (or affective computing, as Picard prefers to name it) only started to gain some momentum after the publication of the influential books of Damasio (Damasio 1994) and Goleman (Goleman 1996).

Some current related work in this field can be found in (Velásquez 1997), where the approach is based on the Minsky's Society of Mind paradigm (Minsky 1988). His work has evolved thereafter to include Damasio (Damasio 1994) ideas (Velásquez 1998a; 1998b; 1998c). Taking a robotic learning approach, Gadanho (Gadanho & Hallam 1998b; 1998a) came up with a model also based on the Damasio work (Damasio 1994). The OCC theory of emotions (Ortony, Clore, & Collins 1988) has also served as an inspiration for several AI models of emotions, such as the Em module of OZ project (Reilly & Bates 1992; Bates, Loyall, & Reilly 1992), and the TABASCO architecture (Staller & Petta 1998), based on the emotion appraisal theory. On the side of affective computing (Picard 1997), *i.e.,* the human-machine interaction on an emotional basis, Picard (Picard 1995; Vyzas & Picard 1998) and Cañamero (Cañamero 1997) have developed some interesting research paths.

There are several aspects that distinguish the model developed by the authors and other approaches. On the one hand, the model is oriented towards the emergence of artificial emotional behavior from a particular architecture, without an *a priori* definition of human-like emotions. There are a reason for this: the objective of this research is not explaining human emotions (and feelings, in the Damasio definition (Damasio 1994)) but rather creating a theoretical and abstract framework uncompromised with human emotions. On the other hand, besides and beyond mimicking emotional behavior, this approach aims at covering more generic aspects of intelligence, such as primordial meaning (Ventura & Pinto-Ferreira 1998b), relevance assessment (Ventura, Custódio, & Pinto-Ferreira 1998a), and decision making under partial ignorance. A final concern also includes efficient response to the environment (Ventura & Pinto-Ferreira 1998a).

In the following section, the foundations of the model are briefly described. In section 3, the model is presented, which will be used in section 4. This papers ends with some conclusions and related work.

## Foundations

The intelligence that distinguishes humans from other mammals is related with cognitive functions, such as reasoning, planning, and so on. These abilities are commonly associated with the neocortex. But according to Damasio, even these higher cognitive abilities use emotions to function properly (Damasio 1994).

To explain the role of emotions in rationality, Damasio raises the *somatic marker* hypothesis (Damasio 1994): certain experienced events left an association between the perceptual stimulus[1] and the response it elicited in the body (e.g., gut feeling). In other words, some images perceived during the event became *marked* with a representation of the body state at that time. Furthermore, the establishment of somatic marks may not require the actual presence of stimuli eliciting a body state. Previous somatic marks can be *propagated* through associations, for instance. The way this propagation occurs is not explicit in Damasio literature (Damasio 1994). While it is essential to a prescriptive model, it may be considered secondary for a descriptive one.

## The Model

The model presented in this paper is based on a double-representation paradigm previously discussed in (Ventura & Pinto-Ferreira 1998a; Ventura, Custódio, & Pinto-Ferreira 1998b; 1998a). It is hypothesized that stimuli are processed under two different perspectives. The first one extracts a *cognitive image* aimed at pattern matching and it is rich enough to allow a fairly good reconstruction of the original stimulus, and the second one creates a *perceptual image* that is a simple, small, reduced set of essential features which are "meaningful" to the agent in the sense that they form the built-in substratum (e.g., a vector of features like size, fast movement, quick approach, dominant color, etc.). This double representation spawns a major division of the model into a cognitive and a perceptual layer (see figure 1). It is important to stress that although the term "perceptual" is being assigned to the perceptual layer, both layers do respond to perceived stimuli. The use of the term "perceptual" aims at distinguishing it from the cognitive kind of processing. The perceptual processing is centered on a small set of basic features extracted from an input stimulus.

Besides the perceptual image, there is another representation in the perceptual layer termed *Desirability Vector* (DV for short). Each one of the DV components represents a basic kind of assessment of a stimulus. Each component can be either activated or neutral (varying either discretely or continuously). Neutral components mean no assessment. But when a certain component is activated, it means that the stimulus triggers a specific basic assessment, e.g., is it good? is it bad? Certain basic stimuli are able to trigger, at a first level, certain components of the DV. For instance, a threatening stimulus, may activate a "fear" DV component, which ultimately generates a fear behavior.

Here follows a summary of how the model works: in response to an external stimulus, the cognitive and the perceptual layer process it in parallel. At the perceptual layer, there is a direct map between stimuli and the DV.

[1]Also termed *image* in this paper, including not only visual images, but also information originating from other sensors, such as auditory, tactile, and so on.

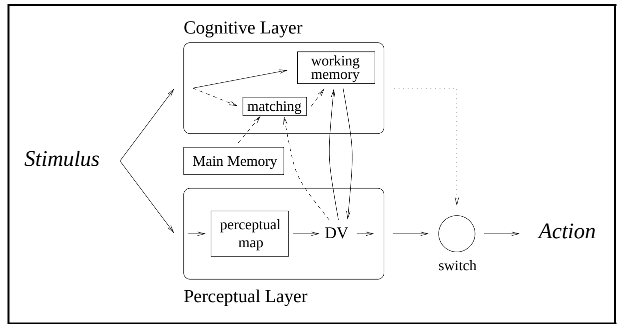


Figure 1: The complete picture of the proposed model, containing all the components discussed in the above sections.

When the agent is built, a part of this mapping must already exist, in order to allow it to bootstrap. Furthermore, this map is able to be adaptive. This forms a kind of implicit memory, termed *perceptual memory.* On the other hand, the cognitive processor looks for matches in the main memory. This memory contains experienced associations, but unlike the perceptual memory, these associations are individually stored as representing events[2]. These associations contain both the cognitive image, the corresponding DV, and the perceptual image. The origin of this DV comes primarily from the perceptual layer, but one can also consider propagating DV instances from other associations. This is a way to allow the agent to associate cognitive images to DV instances, even when faced with a situation where the input stimulus does not deliver (in the perceptual mapping) a significant DV. This memory is here termed *main memory.* The *working memory* holds the input cognitive image, the DV (and optionally the perceptual image), as well as the results from the matching process (or any other higher-level cognitive processes). The action, in response to the stimulus (if any) comes primarily from the DV, although there is provision for actions originating from the cognitive layer. If the agent decides on any action, it may produce alterations in the environment, which can be perceived by the agent as a *feedback stimulus.* This new stimulus tells the agent the result of its action. It is fed into the architecture, in order to make the agent learn. This learning can be accomplished at several levels: at the perceptual layer, it can adapt the perceptual map to be sensible to new stimuli, and at the cognitive layer, it can mark (one or more) cognitive images with the DV, along with the action that led to the environment feedback.

## Implementation

This section describes three implementations of the proposed model. The common ground for the implementations is an episodic environment. Each episode starts with a stimulus applied to the agent, followed by the

[2]Note that in the future, other kinds of representations other than events may take place in this memory.

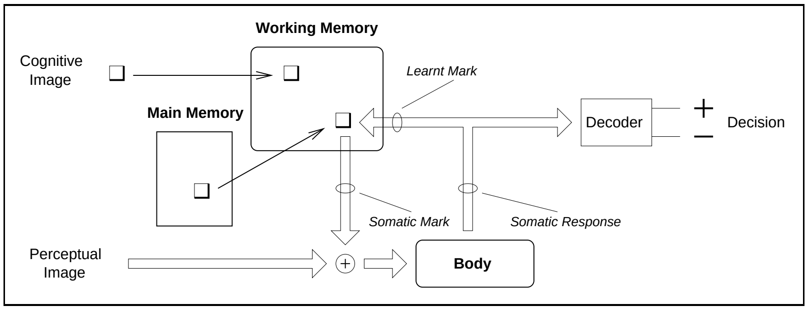


Figure 2: Architecture of the `damasio` implementation.

agent decision/action, and possibly a response from the environment in the form of a second stimulus.

## The `damasio` implementation

The first implementation is called `damasio` and it aims at experimenting the marking mechanism described in the previous section. To understand the basic idea, picture a person being frightened by the occurrence of a thunder. After a quick flash of light, (s)he is stricken by the scary sound of a thunder. Assume that the group lightning and thunder can be considered as a single, complex stimulus because of the short time lapse separating these two events. Furthermore, assume that the lightning is far away from the observer, and as consequence, it only has a relevant cognitive image (low perceptual relevance) whereas the thunder contains a strong perceptual impression (because of its intensity). Of course each of these aspects of the complex stimulus can be considered as having both cognitive and perceptual parts. According to the model, (s)he then forms an association between the flash of light (cognitive part) and the thunder (perceptual part). Then, every time (s)he senses the flash of light alone, the memory of a thunder is recalled, indicating the scary nature of the stimulus.

In this implementation, stimuli are delivered to the agent pairwise: a cognitive stimulus and a perceptual stimulus, both implemented using bidimensional real vectors. The rationale is to view the absolute position of the vector in the 2D real plane as a cognitive representation, and to interpret the perceptual vector in an hardwired fashion: the first and second components denote "amounts" of positiveness and negativeness of the stimulus. For instance, the stimulus pair $<(2,4);(0.9,0)>$ denotes a very positive stimulus (considering the perceptual vector components ranging from 0 to 1) consisting of a point in the plane with coordinates $(2,4)$.

The architecture of this implementation is shown in figure 2. The agent perceives external stimuli through two channels: the cognitive part of the stimulus (e.g., shape of the lightning), and the perceptual one (e.g., the thunder). There is a (short-term) working memory, where the present input is used to recall past associations, and an output is obtained; and a (long-term) main memory, where associations are stored throughout the agent life. The recalled associations are combined with the environment input to derive a body response (labeled "somatic mark"). This body response (labeled "somatic response") is used to trigger a decision (positive or negative, for simplicity — "is it good?" or "is it bad?"), and to update the association, depending on its similitude to the stimulus.

The system works as follows: each stimulus corresponds to a pair (cognitive, perceptual) of vectors. The cognitive vector is copied into the working memory, and the main memory is browsed for similar vectors. For simplicity, all associations from the main memory are considered, but only a pre-defined number of the most similar ones are chosen and copied to the working memory. In the working memory, these associations form *frames*. A frame contains the recalled association (the cognitive vector and a mark vector), and the similarity measure. Next, each of these frames are combined with the perceptual input. Figure 3 shows this mechanism in detail.

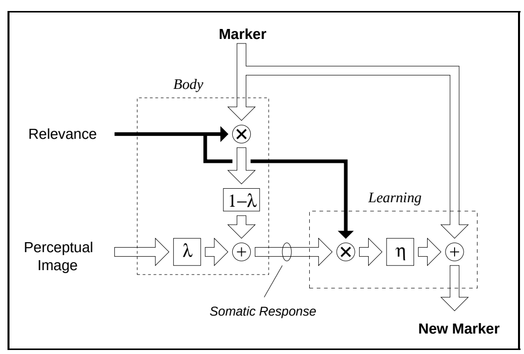


Figure 3: Marking mechanism in the `damasio` implementation. A body response ("somatic response") and an updated mark is computed, from the perceptual input, the old mark, and a similarity measure.

Using the perceptual image, the mark, and the similarity measure (termed "relevance"), a body ("somatic") response and an updated mark are computed. This mark is associated to the originating association, and supersedes the corresponding association in the main memory. Note that the incoming stimulus always forms a new frame in the working memory, and its mark is initially put to zero (null vector), and the similarity measure put to 1 (maximum similarity). For each frame $F_n$ ($n = 1, 2, \ldots$), these operations are performed according to the formulas

$$R_n = \lambda I_P + (1-\lambda) s_n M_n \qquad (1)$$

$$M'_n = \eta s_n R_n \qquad (2)$$

where $I_P$ stands for the perceptual image, $M_n$ and $s_n$ for the frame mark and its similarity measure, $R_n$ the body response, and $M'_n$ for the updated mark value. The rationale behind equation (1) is to linearly interpolate between the present perceptual image and the body response marked on the recalled image, weighted by the similarity measure $s$ (relevance), which ranges from 0 (not similar at all) and 1 (maximum similarity). This interpolation is controlled by the $\lambda$ coefficient ($0 \le \lambda \le 1$). The role of $s_n$ is to allow the recalled mark to influence the outcoming somatic response $R_n$, depending on the similarity found between the present stimulus and the recalled one. Strong marks on very similar stimulus should elicit higher body responses than less similar ones. This similarity measure $s_n$ accounts not only for the cognitive image similarities, but also for the perceptual image. With respect to (2), the idea is to update the new mark $M'_n$ according to two coefficients: the similarity measure (the more similar the stimulus is, the more it should be updated), and a learning rate $\eta$.

As it was previously noted, both the cognitive and perceptual images are bidimensional vectors, as well as the referred marks. The similarity measure is evaluated using the following expression:

$$d(u,v) = \exp\left[t\sqrt{(u_1-u_2)^2 + (v_1-v_2)^2}\right] \qquad (3)$$

where $u = (u_1, u_2)$ and $v = (v_1, v_2)$ are the considered images. The constant $t < 0$ conditions the decay rate as $u$ and $v$ become apart. This constant can be interpreted as a tolerance value — "how much shall I consider this (non-identical) image pair similar?". The expression used for measuring mark similarities is the same. The total similarity, between the stimulus and the recalled frame is weighted by $\xi$ ($0 \le \xi \le 1$) between these two measures:

$$s = \xi d(I_C, I^M_{C_n}) + (1-\xi) d(I_P, M_n) \qquad (4)$$

where $I_C$ and ($I^M_{C_n}$) denote the input and the recalled cognitive images.

The experimental setup for this implementation comprises three phases. First, a set of four stimuli was presented, two of them strongly positive, and the other two strongly negative. These stimuli are called *A1-*, *A2+*, *A3+*, and *A4-*. The ending signal is + or - depending on whether they are positive or negative. The location of the stimuli in the Cartesian plane is shown in figure 4 as bullets. Positive stimuli have perceptual image $I_P = (0.8, 0)$ while the negative ones have $I_P = (0, 0.8)$. The agent was sequentially stimulated with this set of four stimuli four times, in order to get them clearly marked in the agent memory.

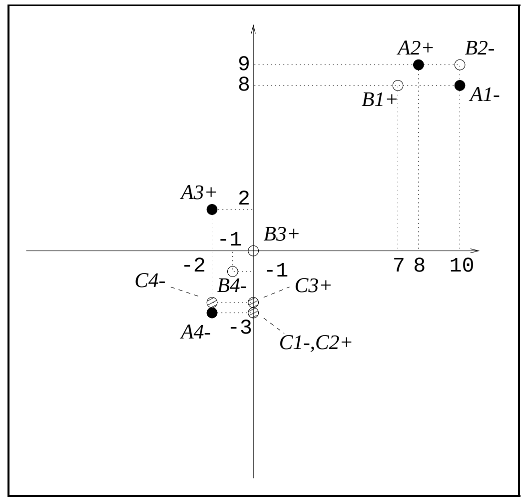


Figure 4: Location of the stimulus cognitive image vectors in the `damasio` experiment. See text for the experiment description, as well as the used notation.

Next, a series of four stimuli with null perceptual image $I_P = (0, 0)$ were applied. These stimuli are denoted *B1+*, *B2-*, *B3+*, and *B4-*, where the signal now represents the agent assessment, *i.e.*, whether the strongest classification is positive or negative. As expected, these results are consistent with the closest stimuli experienced in the first phase. This shows that after the agent being submitted to a set of "strong" stimuli, it learnt, and when stimulated with null perceptual image stimuli, the agent was able to classify them according to its previous experience. The output of the implementation can be seen below, where for each stimulus, the "strongest" frame $n$ is shown:

| $I_C$ | $I_P$ | $I^M_{C_n}$ | $s_n$ | $R_n$ | D |
|---|---|---|---|---|---|
| (7,8) | (0,0) | (8,9) | 0.394 | (0.254,0) | + |
| (10,9) | (0,0) | (10,8) | 0.494 | (0,0.319) | – |
| (0,0) | (0,0) | (-2,2) | 0.247 | (0.160,0) | + |
| (-1,-1) | (0,0) | (-2,-3) | 0.286 | (0,0.184) | – |

Finally, an experiment to test the expert discrimination capability of the agent. A stimulus *C1-* with null perceptual image was applied, and as expected, the agent answered with a negative assessment (closest to *A4-*). Then, a positively marked stimulus *C2+* was applied ($I_P = (0.8, 0)$). Two "colorless" ($I_P = (0, 0)$) stimuli, *C3+* and *C4-*, were applied, resulting in a positive to the first and negative to the second. Given a new scenario with the new stimulus *C2+*, the agent answered coherently, showing its ability to discriminate between *C3+* and *C4-*:

| $I_C$ | $I_P$ | $I^M_{C_n}$ | $s_n$ | $R_n$ | D |
|---|---|---|---|---|---|
| (0,-3) | (0,0) | (-2,-3) | 0.308 | (0,0.173) | – |
| (0,-3) | (.8,0) | (0,-3) | 1 | (0.8,0) | + |
| (0,-2.5) | (0,0) | (0,-3) | 0.685 | (0.384,0) | + |
| (-2,-2.5) | (0,0) | (-2,-3) | 0.685 | (0,0.332) | – |

These experiments were performed setting the parameters $\lambda = 0.3$, $\eta = 1$, and the working memory was limited to 5 frames. These constants condition the behavior of the agent in ways that allow some interesting considerations on possible interpretations. For instance, taking the $\lambda$ parameter, which interpolates the somatic response between the perceptual image and the recalled mark, when significantly reduced (say, $\lambda = 0.05$), makes the agent less sensible to the perceptual image, relying more on its past experience than in present reality. Consider that right after the initial sequence of stimuli *A1* to *A4,* is applied a stimulus with cognitive image $(10, 9)$ (same as *B2*) and perceptual image set to $(0.4, 0)$ (mild positive). With $\lambda = 0.3$ the agent accepts the new stimulus, assigning a positive classification (it disregards the "negative experience" of *A1-*):

| $I_C$ | $I_P$ | $I^M_{C_n}$ | $s_n$ | $R_n$ | D |
|---|---|---|---|---|---|
| (10,9) | (0.4,0) | (10,9) | 1 | (0.4,0) | + |

But when the $\lambda$ parameter is reduced to 0.05, the agent disregards now the positive perceptual image, assessing the stimulus as negative (due to the influence of *A1-*):

| $I_C$ | $I_P$ | $I^M_{C_n}$ | $s_n$ | $R_n$ | D |
|---|---|---|---|---|---|
| (10,9) | (0.4,0) | (10,8) | 0.494 | (0.020,0.433) | – |

How can this behavior be interpreted? The $\lambda$ parameter plays an interesting role of making the agent more or less trustful of the perceptual, when faced with a contradictory past experience. This result has some similarity with a "superstitious" behavior.

This implementation deals only with the marking mechanism. The stimuli are very basic, not reflecting the complex nature of the cognitive memory. Furthermore, there is no action (and consequently no perceptual feedback). Associations are always done, filling the agent memory with data that may not be relevant. But the results are interesting, in the sense of showing the marking and the memory retrieval mechanisms.

## The `faces` Implementation

This implementation presents several sophistication over the preceding one. The objective is to experiment with more complex stimuli models, as well as the environment feedback. So, the stimuli (equal to the cognitive images) are a square set of polychromatic pixels (16 by 16). The mapping between the stimulus and the DV is fixed by design. In fact, the perceptual map discussed in the section  used the perceptual image as an intermediate representation. This perceptual image contains a set of basic features extracted from the stimulus. These features are then mapped into the DV. Both maps are hard-wired.

The agent perception of the environment is limited to the 16 by 16 pixel images. Each pixel is one of `blank` (background), `black`, `green`, or `red`. The agent can have one of three decisions: `none` (inaction), `accept`, or `reject`. The perceptual features extracted are: number of red pixels (assessment of "redness"), number or green pixels (assessment of "greenness"), and total number of non-blank pixels (measure of object size). The DV has three components: three boolean components, indicating whether or not the stimulus is "good," "bad," or "deadly" (*i.e.,* very dangerous). The perceptual image is mapped into the DV using a set of thresholds. For instance, if the total number of pixels is above a predetermined threshold, and the number of green pixels is above another threshold, the "good" components of the DV is activated. In this implementation, the presence of green pixels corresponds to a "good" stimulus, while red pixels denote a "bad" one.

The model of this implementation is depicted in figure 5. The cognitive layer uses both the cognitive and the perceptual images to search for a memory match. The perceptual image is first used to select a limited set of candidate memory associations (termed memory frames). Note that this is an implementation of an indexing mechanism raised in the section . From those, the cognitive image selects the best match. If three conditions hold, the frame action is selected. Otherwise, the direct perceptual path is used to derive the action. These conditions are: there is a match, the difference measure between the cognitive image and the memory frame is below a certain threshold. This difference measure is simply the Hamming distance between the two images[3].

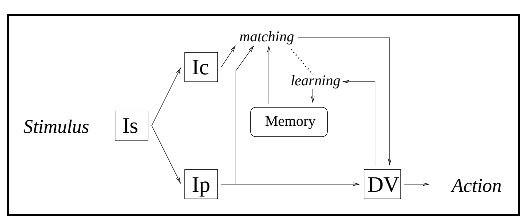


Figure 5: Architecture of the `faces` implementation.

A memory frame contains the cognitive and perceptual images, the DV, and an action list. This list consists of pairs (action, future frame), and is used to de-

[3]The Hamming distance is the number of pixels differing between the two images. See (Ullman 1996) for a definition and related issues.

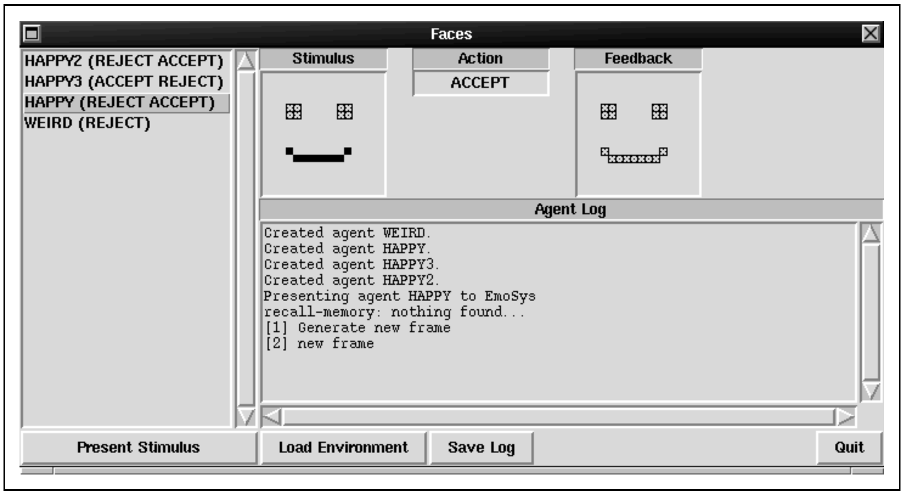


Figure 6: Screenshot of the `faces` implementation: a smiling face with some green pixels.

cide on the next action, based on the past experience. When a memory frame is selected as a match for the current stimulus, its action list is browsed, and the action that leads to the most favorable scenario is chosen. Each scenario is evaluated according to its DV (the positive component means +1, the negative -1, and the "deadly" -10; the heuristic to be minimized is the sum of the values of the corresponding activated components). If no match is found, or there is no action list, the agent acts accordingly to a built-in DV action map (negative or "deadly" leads to a `reject`, positive to an `accept`, and `none` otherwise).

After the agent action, the feedback stimulus is applied to the system, and the resulting memory frame is stored in the main memory. Furthermore, the action list of the original stimulus frame (before the action be performed) is updated/set, pointing to the feedback frame. Next time the agent faces a similar situation where this frame is recalled, it will know what to expect from the corresponding action.

An illustrative experiment will be presented below, consisting of a sequence of stimuli. In the following screenshots, green pixels are denoted by (⊞), and red pixels by (▩). Prior to the agent first stimulus, the memory is blank. The first stimulus (figure 6) consists in a smiling face silhouette with some green pixels (a perceptual positive DV). The agent uses the perceptual assessment indicating an `accept` action. The environment responds with a all-green face (*i.e.,* positive DV). The corresponding association is formed and stored in memory.

Next, a colorless face, which is similar to the first one, is presented (figure 7). The agent recalls the previous association, and chooses to `accept` the stimulus. However, if this stimulus were presented without the former association, the action would be `none` — the stimulus would be mapped by the perceptual layer to a null DV.

An interesting result is obtained when now, a similar face is shown, containing some red pixels (figure 8). In this case, the recalled association is used to *override* the perceptual impulse to `reject` the stimulus, so a `accepting` it. This case illustrates the role of the cognitive layer in providing a refined response, than the basic perceptual one. Using the same line of reasoning, if this stimulus were shown prior to the first of the sequence, the agent would `reject` it.

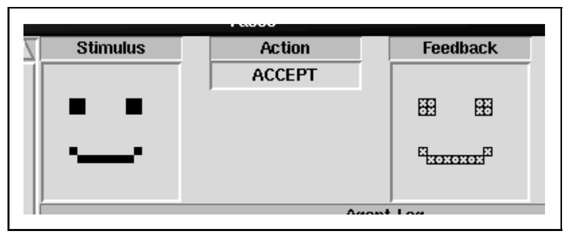


Figure 7: Screenshot of the `faces` implementation: a similar smiling face all in black.

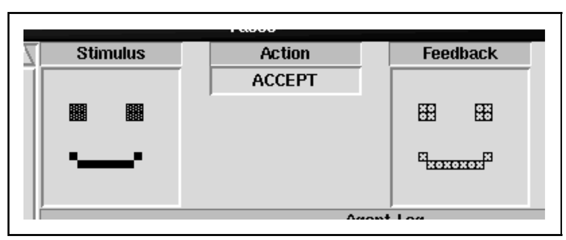


Figure 8: Screenshot of the `faces` implementation: a similar smiling face but with some red pixels (the "eyes").

At last, a different face is shown (with some red pixels, figure 9), and unlike the previous stimulus, because this face is "unknown" to the cognitive layer, the action is `reject`, following the perceptual negative assessment.

Other experiments were performed with the archi-

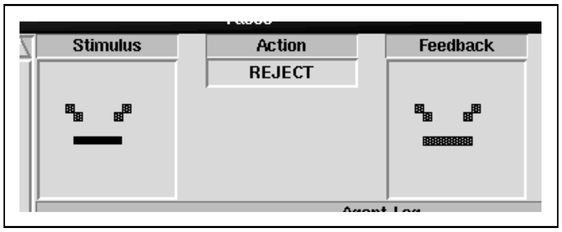


Figure 9: Screenshot of the `faces` implementation: a distinct face with some red pixels.

tecture, showing another interesting results. For instance, if the acceptance of the stimulus of the figure 6 had a negative response (e.g., a very "red" face), next time that same stimulus was presented, the agent would `reject` it. When the action resulting from a given stimulus is answered with a negative response, the agent will not repeat the mistake — other actions are "tried" in a seek for a better response. The frame that this action points to has a negative DV, making the agent to avoid it.

It is very clear in this implementation, the role of the built-in knowledge. The mechanism that is behind the agent behavior facing environment stimuli, is encoded in the perceptual layer. Namely in the perceptual mapping between stimuli and the DV. It is on top this layer that the cognitive layer works. When the simplicity of the perceptual layer is not enough to cope with a complex environment, the cognitive one jumps in, providing the "knowledge" gained from past experience.

## The `decks` Implementation

The objective of the `decks` implementation is to reproduce the results of the deck game (figure 10) described by Damasio ((Damasio 1994) page 212), using the proposed model. This game consists of four decks — A through D. The subject is asked to turn a card, from a deck of her/his choice, then the (s)he is told whether that card made her/him lose or gain a certain amount of (fake) money (from a start loan of $2,000).

In a simplified version of the original game (Bechara *et al.* 1994; 1997), decks A and B usually give $100 except for a few cards that make the player lose -$1250, while decks C and D usually give a lower value of $50 where there are more frequent losses of -$250. The net profit of decks A and B is negative, while decks C and D provide a positive one.

In the original experiment (Damasio 1994) normal people usually started the game trying each one of the decks, but soon after taking note of the high losses resulted from the A and B decks, they converged taking cards only from decks C and D. However, patients with prefrontal lobes lesions, kept on taking cards from the apparently more profitable decks A and B, insensitive to the high losses that, now and then, cards from those decks undertook (figure 10). These patients were unable to recall the risk of choosing A or B deck cards (*i.e.,* its somatic marker), and kept on choosing the immediate higher value of these decks. Damasio called this phenomenon "myopia for the future" (Damasio 1994).

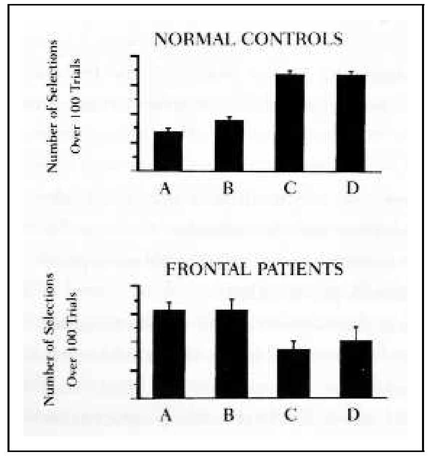


Figure 10: Number of selections from each of the decks, in normal subjects and frontal patients. (From (Damasio 1994) page 215, reprint by courtesy of the author.)

In terms of the implementation, the environment is episodic, with an environment feedback phase. First, four stimuli are simultaneously presented to the agent (four symbols, corresponding to the four decks: *A, B, C,* and *D*; if a deck is empty, it is not presented to the agent). The agent action is simply the choice of a deck. The environment responds with the amount of money gained/lost. Each stimulus encompasses a pair of symbol and money amount gained (negative, if lost). In the first phase, the second components of all stimuli are null (the card amount is obviously hidden). Only after the action the reward associated with the chosen card is revealed. The perceptual layer only extracts the money amount (the perceptual image), while the cognitive layer extracts the symbol. The DV has only two (boolean) components, one for positive and other for negative assessment of the deck. The mapping between the perceptual image and the DV activates the positive component if the amount greater than zero, or the negative component when it is less than zero.

The model of this implementation is identical the one represented in figure 1. An important innovation facing the previous two implementations is the adaptability of the perceptual layer. Both kinds of learning are implemented: the cognitive event-based learning, and the perceptual mapping-based learning. When the agent is faced with the four decks, the perceptual layer is able to give an immediate assessment of the desirability of each deck, while the cognitive layer browses the memory for past events associated with each deck. With all this information in the working memory, the agent decides which deck to choose.

The working memory is organized in clusters of frames. Each cluster corresponds to a specific deck, and contains the input stimulus (the deck symbol only), the perceptual frame (the expected perceptual image and the expected DV, or in other words, the expected amount of gain/loss), and the frames recalled from memory (obtained by the cognitive layer). When each frame is complete, a representative frame is chosen for each cluster. Then, all the clusters with a negative DV are rejected, and a deck is randomly chosen from the remaining ones. In fact, the perceptual value is used to weight this random choice, in order to make the agent prefer higher value cards. But if all clusters are rejected, then the action is randomly chosen from all the available decks, also using a weight factor.

After choosing the deck, the environment responds with a feedback stimulus, now containing not only the symbol of the deck, but also the amount of money gained/lost. This information is used to update the perceptual map (according to a learning rate), and to add the frame to the main memory, associating the cognitive and the perceptual images, along with the DV (mapped from the perceptual image, *i.e.,* the amount of money). This perceptual image can be interpreted here as the expected gain. In the perceptual layer learning, the update rule of this expected value is simply:

$$V'_m = \eta V_p + (1 - \eta) V_m \tag{5}$$

where the new memory frame expected value $V'_m$ is interpolated between its former value $V_m$ and the feedback value $V_p$, using the learning rate $\eta$.

In order to simulate the behavior of the frontal (abnormal) patients playing this game, the agent was prevented from recalling memory frames. Then, the perceptual layer was left alone to decide which deck to choose, preferring the decks A and B, because of the most frequent \$100 cards. As an example, setting the learning rate parameter to $\eta = 0.001$, the obtained results, shown in figure 11, are similar with the Damasio experiments results of figure 10.

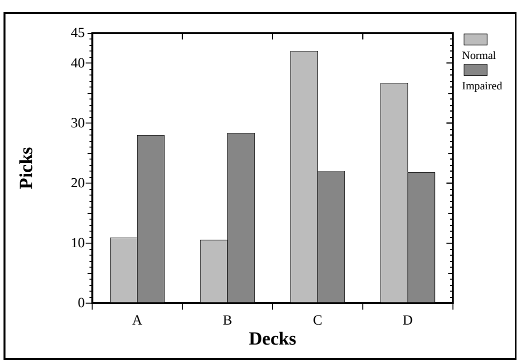


Figure 11: Results from the `decks` implementation. The average number of picks for each deck is shown. The average was taken over 200 experiments of 100 turns each. The $\eta$ parameter was set to 0.001.

These results illustrate the distinct natures of the learning process performed by each layer. But they are not to be considered separately. Although the perceptual layer is able to work by itself, the same cannot be said about the cognitive layer. This is because the cognitive layer uses the perceptual representation, in order to contribute to an overall enriched behavior.

## Conclusions and Future Work

One of the interesting results got from the implementation of the theoretical model is that the developed agents exhibit a behavior that can be seen as "emotional." This assertion deserves some explanation. It was assumed by the authors that the endeavor of formally defining the concept of emotion is not worth pursuing — at least in what concerns "artificial emotions." (For instance, it is a waste of time to create a definition of intelligence in order to explain whether an agent exhibits artificial intelligence or not). From a behavioral point of view, all the three implementations show the ability of dealing with unpredictable stimuli, making adequate decisions efficiently, *i.e.,* without the need of wasting time in exhaustively analyzing the cognitive aspect of the stimuli. Recall that efficiency is a characteristic of emotional decision systems (Damasio 1994). This quick response to the environment should not be confused with the one exhibited by reactive systems. Although the latter uses the environment as the sole representational mechanism, the former is able to build its own associations to learn with the environment. Other characteristic which can be associated with emotional decision making is the ability of constructing a certain kind of meaning bootstrapped on top of a basic set of built-in associations. This basic associations were essential to ensure the adequacy of the implemented agent decision making process. For example, recall that in the `decks` implementation the agent decided based on the "semantics" of the perceptual image previously associated with losses and gains.

It is important to note that this model is still unable to explain certain "high-level" emotions, such as shame and guilt. The rationale behind the development of this model is to first cover the most basic aspects of emotions, according to Damasio. In the presented model, the DV is directly related with action, which it seems it is not the case with shame or guilt. It is assumed that only after most basic aspects of the model are mastered, one can worry about these "high-level" emotions, which are more indirectly connected with the agent action.

The model presented in this paper along with the described implementations leave open a variety of questions, possibly leading to some interesting research paths. In the last implementation, two distinct kinds of learning were implemented: one in the perceptual layer, which resembles reinforcement learning in the aspect of updating through time a set of parameters according to a feedback from the environment, and another at the cognitive layer, which is basically an instance-based kind of learning (Mitchell 1997). However, these two learning processes are not independent: they work together. It would be useful to study the theoretical implications of having these two learning processes intertwined this way. A second path of research is related with the inclusion of more complex emotions (e.g., guilt, shame) which are indirectly related to the agent action. Finally, it is necessary to move out of the simple episodic environments used in the described implementations, either by applying the model to environments requiring some elaborative abilities, or by experimenting with real robots in the physical world. In either case, some useful lessons can be taken from the work of Piaget (Piaget & Inhelder 1969). His research on child development provide interesting cues on how progressively complex cognitive abilities appear on top of more basic ones.

The ultimate goal of this research is to build agents (namely embodied in physical robots) that are able to cope with the dynamic and complex environments humans live in, learn to interact with them, and to show intelligent behavior in domains for which it was not specifically programmed. Picture for instance teaching a robot to play chess, not by developing search algorithms and fine-tuning heuristics, but by showing it an actual chess board, the pieces, the rules, and allowing it to develop itself by playing, rather than simply programming it in a way that reflects the ideas of the programmers.

## References


Bates, J.; Loyall, A. B.; and Reilly, W. S. 1992. An architecture for action, emotion, and social behavior. In *Proceedings of the Fourth European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, Decentralized AI Series. Elsevier/North Holland.

Bechara, A.; Damasio, A. R.; Damasio, H.; and Anderson, S. W. 1994. Insensitivity to future consequences following damage to human prefrontal cortex. *Cognition* 50:7–15.

Bechara, A.; Damasio, H.; Tranel, D.; and Damasio, A. R. 1997. Decising advantageously before knowing the advantageous strategy. *Science* 275:1293–1295.

Cañamero, D. 1997. Modelling motivations and emotions as a basis for intelligent behavior. In *Proceedings of Agents'97*. ACM.

Damasio, A. R. 1994. *Descartes' Error: Emotion, Reason and the Human Brain*. Picador.

Gadanho, S. C., and Hallam, J. 1998a. Emotion-triggered learning for autonomous robots. In Cañamero, D.; Numaoka, C.; and Petta, P., eds., *Workshop: Grounding Emotions in Adaptive Systems*, 31–36. SAB'98: From Animals to Animats.

Gadanho, S. C., and Hallam, J. 1998b. Exploring the role of emotions in autonomous robot learning. In Cañamero, D., ed., *Emotional and Intelligent: The Tangled Knot of Cognition*, 84–89.

Goleman, D. 1996. *Emotional Intelligence*. Bloomsbury.

Minsky, M. 1988. *The Society of Mind*. Touchstone.

Mitchell, T. M. 1997. *Machine Learning*. McGraw-Hill.

Ortony, A.; Clore, G. L.; and Collins, A. 1988. *The Cognitive Structure of Emotions*. Cambridge, UK: Cambridge University Press.

Piaget, J., and Inhelder, B. 1969. *The Psychology of the Child*. Basic Books, Inc.

Picard, R. W. 1995. Affective computing. Technical Report 321, M.I.T. Media Laboratory; Perceptual Computing Section.

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

Reilly, W. S., and Bates, J. 1992. Building emotional agents. Technical Report CMU-CS-92-143, CMU, School of Computer Science, Carnegie Mellon University.

Sloman, A., and Croucher, M. 1981. Why robots will have emotions. In *Proceedings IJCAI 1981*.

Staller, A., and Petta, P. 1998. Towards a tractable appraisal-based architecture. In Cañamero, D.; Numaoka, C.; and Petta, P., eds., *Workshop: Grounding Emotions in Adaptive Systems*, 56–61. SAB'98: From Animals to Animats.

Ullman, S. 1996. *High-level Vision*. The MIT Press.

Velásquez, J. D. 1997. Modeling emotions and other motivations in synthetic agents. In *Proceedings AAAI-97*, 10–15. AAAI.

Velásquez, J. D. 1998a. A computational framework for emotion-based control. In Cañamero, D.; Numaoka, C.; and Petta, P., eds., *Workshop: Grounding Emotions in Adaptive Systems*, 62–67. SAB'98: From Animals to Animats.

Velásquez, J. D. 1998b. Modeling emotion-based decision-making. In Cañamero, D., ed., *Emotional and Intelligent: The Tangled Knot of Cognition*, 164–169.

Velásquez, J. D. 1998c. When robots weep: Emotional memories and decision-making. In *Proceedings AAAI-98*, 70–75. AAAI.

Ventura, R., and Pinto-Ferreira, C. 1998a. Emotion-based agents. In *Proceedings AAAI-98*, 1204. AAAI.

Ventura, R., and Pinto-Ferreira, C. 1998b. Meaning engines — revisiting the chinese room. In Cañamero, D.; Numaoka, C.; and Petta, P., eds., *Workshop: Grounding Emotions in Adaptive Systems*, 68–70. SAB'98: From Animals to Animats.

Ventura, R.; Custódio, L.; and Pinto-Ferreira, C. 1998a. Artificial emotions — goodbye mr. spock! In *Progress in Artificial Inteligence, Proceedings of IBERAMIA'98, Lisbon, Portugal*, 395–402. Colibri.

Ventura, R.; Custódio, L.; and Pinto-Ferreira, C. 1998b. Emotions — the missing link? In Cañamero, D., ed., *Emotional and Intelligent: The Tangled Knot of Cognition*, 170–175.

Vyzas, E., and Picard, R. W. 1998. Affective pattern classification. In Cañamero, D., ed., *Emotional and Intelligent: The Tangled Knot of Cognition*, 176–182.