EXPLORING *Kansei* IN MULTIMEDIA INFORMATION

Nadia BIANCHI-BERTHOUZE* and Luc BERTHOUZE**

* Aizu University, Tsuruga, Bki-machi, Aizu Wakamatsu 965-8580, Japan  
** Neuroscience Research Institute, AIST Tsukuba, Tsukuba 305-8568, Japan

**Abstract:** With information technologies being increasingly involved in areas such as (online) shopping, entertainment or advertisement, computer systems are bound to be able to process *Kansei* information, i.e. information relevant to users' sensibilities. Rather than modeling the biology of users' sensibilities, we suggest a functional approach by modeling the translation process between different modalities of expression of a same *Kansei* concept. We hypothesize that this translation process can be grounded into the categorization of users's perception, i.e. the extraction of structures in the multimedia information. Because this translation process is intrinsically variable, we propose a computational agent, called K-Agent, able to learn categories in its visual perception and interactively evolve a translation language. The K-Agent consists of three main modules: a multi-features image processing unit, a learning kernel that iteratively constructs the translation language, and a feedback interpreter system which incorporates self-supervision and user feedback to structurally tune the learning kernel. The concept of K-Agent has been evaluated in a real-world application involving user *Kansei*, more specifically, the filtering of images against a given user *Kansei* impression. Our experimental results demonstrate the feasibility of the concept as well as a superior performance compared to manually filtering the output of existing search engines.

**Keywords:** Subjectivity, Dynamical user model, Multimedia information systems, Image retrieval.

1. **INTRODUCTION**

Information systems are increasingly present in areas such as online shopping, advertisement, entertainment or design. Whereas the computer system is traditionally seen as a reliable and fast mean to manipulate chunks of information such as numbers/text/images in a systematic way, it must now deal with *Kansei* information, i.e. user-specific, sensibility-laden information.

In the recent years, some researchers (especially in Japan) have attempted to endow computer systems with the capability to handle subjective impressions [1, 2, 3, 4, 5, 6] in applications such as urban planning, art, design or fashion. In urban planning for example, Shibata [2] proposes a framework to determine the relationship between the visual characteristics of a street landscape and the user impression. Hattori et al. [5] propose a computer-aided art appreciation system where art students can compare their impressions of a painting with those that the painter wanted to convey. In those works, the core idea is to construct (and then use) a sort of psychological profile of the user by mapping subjective impressions onto a more restricted but explicit set of information (e.g. color, texture, shape in the case of visual impressions).

Techniques such as neural networks, genetic algorithm, fuzzy logic or statistical methods are exploited. The user profile consists in a set of relations. When the system is adaptive [3], the participation of the user in the modeling process simply consists in inputting a feedback in the form "relevant" or "non-relevant" without being given the possibility neither to understand the reasons why the model proved inadequate nor to give a more salient feedback [7, 8]. In our experience, such approach typically leads to a learning curve that cannot stabilize, alternatively settling into one of two or more local minima without improving the quality of the system’s performance. Another obstacle to a real improvement of such user model is the implicit perception that the user model is independent of the organization of the low-level information processing. Indeed, in those works, the low-level information processing remains unaltered throughout the construction of the model. In humans however, visual processes are intertwined with other cognitive processes and it is reasonable to believe that with respect to a given unchanged vision processing configuration, a user feedback might be inconsistent over time. In "relevant/non relevant"-type feedback systems, the user isn’t in a position to understand his/her inconsistencies and for ex-
ample shift the focus of attention (an alteration of the information processing). New modalities of interaction must therefore be involved.

2. ISSUES AND APPROACH

Before presenting our framework, we wish to begin with a clear statement about "user model" and "sensibility" (Kansei). Are these two concepts compatible? A "user model" is usually defined as an explicit representation of properties of a particular user. "Sensibilities" refer to the capacity to respond to or to be affected by something, and often, the capacity for intellectual and aesthetic distinctions, feelings, tastes, etc [9]. Hence, when considering sensibility, a straightforward reading of the definition of a "user model" will result in some explicit representation of those emotional capacities of the user. We should dismiss any attempt to model the "biology" of those emotional capacities. If, when observing a street-landscape, visual characteristics are indeed processed by our visual cortex and some subjective experience occur, that subjective experience is also the product of the intertwine-ment of multiple cognitive processes, e.g. memories of past experience, etc. Even though we humans share a similar sensory-motor apparatus and brain structure, we experience a different subjective experience for a same stimuli and even in the course of time, we can ourselves experience differently the same stimuli because of some other factors (e.g. mood, goal orientation). Human cognition is not about monolithic models [7]. Such models only derive from naive models based on subjective observation and introspection and biases from common computational metaphors [10]. Thus, unless we are capable of modeling the entire brain, and "reconstruct" the user's various experiences (sensory, motor, emotive, etc.), there won't be any suitable explicit representation of the sensibilities of the user.

Since we are dismissing (a) the embodied ability to have subjective experience and (b) the cognitive processes by which the user makes use of concepts such as "emotion", we should be asking, which properties of the user should we model?

We claim here that what matters is how subjective experience is communicated. As Frijda and Swagerman [11] noted, *Who cares about his own feelings when they have no consequence, of wanting to approach or avoid or get rid of, be it with regard to external objects or objects of thought? What is interesting in emotion is some relationship to behavior or behavioral intent.* From an application point of view, the goal of a street landscape design system won't be to understand why a user feel relaxed in a particular street arrangement, but how to design a street so that it will gives that feeling of quietness to the user.

Taking insight from human-human communication, the communication of information in general (and subjective information even more) is using multiple channels of communication. Not just words but additional cues such as metaphors, examples, synonyms and other modalities (postural activity, facial expressions, etc.) are used. Even though those cues might relate to different cognitive processes and sensory-motor experience [12], they relate to the same subjective experience (a broader bandwidth of communication). We are observing there a translation process from one modality to the other to ensure successful communication [7]. While two modalities are unlikely to be interchangeable in absolute and one might be richer than another in expressing concepts, they form an interchangeable characterization of the subjective experience of the user. Since the point of this paper is to explore Kansei in multimedia information, we take as a working hypothesis that Kansei information is partially embedded into the low-level features of different expression instances (within or across modality) of a same subjective experience. It is a reasonable hypothesis. A dark image is more likely going to be subjectively experienced as "sad" because its perceptual experience (visual acquisition and processing), in resonance with other cognitive processes such as memory, might recall a "sad" experience (assuming for example that "black" is culturally associated with sad events). Such type of interaction has already been demonstrated. Consider unconscious learning [13] and the Stroop effect [14]. When presented with a list of words written in a variety of colors, performance in a color recognition and articulation task is dependent on the semantic content of the words. The task is very difficult if names of colors are printed in non corresponding colors. If this is agreed, we can then conclude that if a multimedia information system is capable of extracting regularities or any kind of structures from the low-level features of information exchanged with a user in communicating a Kansei experience, then, it is in effect able to
3. IMPRESSION LANGUAGE TRANSLATION

We define an impression as a feeling or emotion experienced following either an external (any sensory-motor perception) or internal stimuli (a memory or any other product of one or many self-induced cognitive processes). Although the implementation discussed later in the paper deals with verbal impression language \( \text{VIL} \) and image impression language \( \text{ILL} \), the system is specified so as to handle any other languages such as sound impression language \( \text{SIL} \). Video impression language would then be seen as an encapsulation of sound impression language and an extension of the image impression language. In this section, we discuss the formalization of the bidirectional translation process \( \text{ILL} \leftrightarrow \text{VIL} \).

The following notations are used throughout the section:

- \( V \) is a set of labels, where a label is defined as a string of characters. The cardinality of \( V \) is \( h \).
- \( I \) is a set of images. Its cardinality is \( g \). Neither \( h \) nor \( g \) are fixed, they can increase at run-time. \( PS \) is the perceptual signature associated to an image of \( I \). The details of its construction are detailed in section 4.1.
- \( D \) is a set of translation functions \( d_i \), where \( i \in 0, ..., h \). Briefly, when fed with a perceptual signature \( PS \), \( d_i \) returns an activation value in \([-1, 1]\).
  The specifics of the implementation of the modules \( d_i \) are detailed in section 4.2.

We define an Image Impression Language \( \text{ILL} \) as a set of Image Impression Words \( \text{IIW} \). A \( \text{IIW} \) is a pair \(< \text{img}, \text{impression} >\) where \( \text{img} \) is an image of \( I \) or any combination of elements of \( I \) and \( \text{impression} \) is a \( \text{Kansei} \) impression conveyed by the sentence and given by the user. Similarly, we define a Verbal Impression Language \( \text{VIL} \) as a set of Verbal Impression Words \( \text{VIW} \). A \( \text{VIW} \) is a pair \(< \text{v}, \text{impression} >\) where \( \text{v} \) is a label of \( V \) or any combination of elements of \( V \) and \( \text{impression} \) is defined as above.

The translation or mapping between the two languages transforms one word from one language into one or many corresponding words in another language, such that impressions associated with the set of new words are either equal to or include the impression associated with the original word. The translation process is performed by two transformations:

\[
\text{verbalization : } IIW \rightarrow_{\text{thr}} \{ VIW_1, ..., VIW_i \}
\]

\[
\text{visualization : } VIW \rightarrow_{\text{thr}} \{ IIW_1, ..., IIW_j \}
\]

where \( \text{thr} \) is an arbitrarily chosen threshold parameter determining the accuracy of the translation process.

---

**Table 1:** Pseudo-code of the \textit{verbalization} function.

The returned \( SL \) contains a list of labels whose associated verbal impressions in \( \text{VIS} \) will match the impression conveyed by the input image \( \text{img} \).

```plaintext
LabelVector Verbalization(Image img) {
    PS = computePerceptualSignature(img);
    SL = null;
    for (i=0;i<h;i++) {
        di = selectModule(i);
        li = di.getLabel();
        activation = di(PS);
        if (activation>thr) then L.addLabel(li);
    }
    return(SL);
}
```

**Table 2:** Pseudo-code of the \textit{visualization} function.

The returned \( SL \) contains a list of images that possibly reflect the impression associated with the verbal impression conveyed by the input label \( v \).

```plaintext
ImageVector Visualization(Label v) {
    SL = null;
    di = selectModule(v);
    for (t=0;t<g;t++) {
        activation = 0;
        PSi = computePerceptualSignature(Ii);
        activation += di(PSi);
        if (activation>thr) then L.addImage(Ii);
    }
    return(SL);
}
```

As reflected by its pseudo-code, the \textit{visualization} function makes use of the same pathway (\text{stimulus} \rightarrow
perceptual processing \rightarrow verbalization) than used by the verbalization function. While it simplifies the implementation of the translation function, is also reflects (at some abstract level) the idea that, like in humans, a word (in the verbal language) gives rise to a mental image, which is then perceptually processed using the same neural pathway than for an external stimulus, the pre-selection of a module acting then as attentional mechanism. Whether such sequence of events is biologically plausible or not has not been investigated yet, however experiments such as the Stroop effect tend to show that such process is plausible.

The translation process between two words $IIW_i = < img_i, imp_i >$ and $VIW_i = < v_j, imp_j >$ will be:

- **complete** if $imp_i = imp_j$,
- **partial** if $imp_i$ and $imp_j$ partially overlap,
- **incorrect** otherwise.

Because impressions in both $IIW$ and $VIW$ are assigned by the user, the completeness of the translation can only be evaluated by the user. That evaluation process being user-dependent, hence non necessarily consistent over time as discussed in section 2.3, it casts strong constraints of adaptability on the translation functions (section 4.2.1).

4. **K-AGENT**

We define a K-Agent as a computational agent capable of interacting and dialoging with a user in order to learn the translation process discussed in Section 3. The K-Agent follows a learning framework defined by: (1) a bootstrap of the learning phase by a basic user profile with explicit information such as gender, age, sex, etc., and (2) a continuous mixed supervised/self-supervised learning phase with both global and local adaptation mechanisms. As depicted in Figure 1, the K-Agent implements such framework with three main components: an image processing kernel (more generally, a sensory-motor processing kernel), a learning kernel and a feedback processing unit which we detail in the following subsections.

4.1. **Image processing kernel**

The role of the image processing kernel is to construct a multi-dimensional representation (called a perceptual signature) of an image in terms of its information of color, texture and shape (see Figure 2). The dimension of this signature is lower than if the whole image were considered however the interest of pre-attentive processing is two-fold: (1) it gives the system some similarity with the human visual processing and (2) it makes post-processing computationally tractable. Our image processing unit integrates several image analysis processes described below.

![Segmentation Space](image1.png)

Figure 2: Information flow in the image processing kernel.

![HSB and SB segmentation spaces](image2.png)

Figure 3: HSB and SB segmentation spaces.

Color processing is based on a HSB perceptual model (hue, brightness and saturation). While hue denotes color, saturation indicates the richness or vibrancy of the color and brightness relates to the intensity of the light illuminating the object. As proposed by Munsell [15], the HSB space is represented as a cylinder in which brightness is the vertical axis, saturation...
the radius and hue the angular displacement (see Figure 3(top)). The hue dimension is divided into 10 sectors, standing for the 5 principal hues (red, yellow, green, blue and violet) and the 5 intermediate hues (orange, yellow-green, cyan, purple and magenta). This number of sectors is an arbitrarily chosen parameter which can be fine tuned according to whether color nuances must be precisely considered. The second and third dimensions of the HSB model are not totally independent since the domain of chromaticity decreases as the value of brightness reaches extremas (white and black). Thus, we consider both dimensions in a two-dimensional space $SB$ which we split into 11 basic areas (black, dark gray, light gray, white, deep, dark, dull, strong, pale, bright, vivid) as shown in Figure 3(bottom).

For each image, the image processing kernel produces two descriptions (called signatures in this paper). The tone signature (11 parameters) is an histogram of the SB space. In order to construct the color signature, each tone of a hue in the image is considered (i.e. 11*10 parameters). However, since hue is hardly detectable by our visual system when saturation is very low (i.e. left area of the SB space), low-saturation tones such as (white, black, etc.) are considered separately. Thus, the total number of parameters for the color signature is reduced to $74 = 7 \times 10 + 4$ parameters.

Shape processing is achieved using Merelli [16]’s algorithm. Region contours are extracted using change of direction of lines of pixels. Since this algorithm is applicable to binary images only, it is applied on a smoothened binarization of each of the 10 hues and the white, black, dark and light gray ranges. Extracted regions are then classified according to their area into three types of class (small, medium and large). A shape signature is computed as follows: for each class, cardinality, color homogeneity (i.e. a measure of uniformity of the color in the region) and average contrast (contrast between a region and its neighbors) are computed. The three biggest regions are also described in terms of hue, tone, length (i.e. the maximum distance between two pixels of a region), area, position, direction, color homogeneity and contrast. 36 parameters form the shape signature.
Texture analysis is performed on partially overlapping sub-regions (the size of which is arbitrarily set to 40 × 40 for 100 × 100 images) of each dimension of the HSB image. The number of regions varies with the dimension of the image (e.g., 9 regions in a thumbnail image, see Figure 2). The parameters extracted are measures of contrast and entropy of the image [17]. Entropy is a measure of the randomness of the relative distribution of pixel values. Higher entropy indicates that pixel values in the image are very similar. Contrast (∑i ∑j (i - j)² ci,j) and entropy (∑i ∑j ci,j log ci,j) are computed on the gray-level co-occurrence matrix of each component of the HSB image. The texture signature consists of 18 parameters.

Using the algorithm described in Table 3, the image processing kernel eventually produces 4 vectors (hue, tone, shape, texture) which form the perceptual signature (PS) of the images for a total of 139 parameters.

```
PS computePerceptualSignature(Image I) {
    mat1 = RGBtoHSB(I.mat);
    segmat = segmentationHSB(mat1);
    hsbSig = computeHSBSignature(segmat);
    toneSig = computeToneSignature(segmat);
    for(i=0; i<10; i++) {
        binmat = createBinaryImage(segmat, i);
        regions = extractRegions(binmat);
        shapeSig[i] = computeShapeSignature(regions);
    }
    segmat = RGBtoB(I.mat);
    textureSig = computeTexture(segmat);
    // Create PerceptualSignature PS
    PS.add(hsbSig);
    PS.add(toneSig);
    PS.add(shapeSig);
    PS.add(textureSig);
    return(PS);
}
```

Table 3: Algorithm for PS perceptual signature.

4.2. **Word-modules dᵢ**

Word-modules dᵢ, which form the learning kernel, implement the translation functions discussed in section 3. Depicted in the upper-left area of Figure 1, they learn the mapping from a perceptual signature (in our implementation, the 4 perceptual vectors produced by the image processing kernel) onto an activation value for the associated word. A word-module consists of:

- a label lᵢ ∈ V,
- a training set of positive (resp. negative) samples, i.e. images which the user evaluated as relevant (resp. irrelevant) to the word-module,
- a set of neural networks nᵢ,j, where j ∈ {0, 3}, i.e. one neural network for each perceptual channel. Each network is associated with a saliency value sᵢ,j. The functional equation of the word module dᵢ is given by: activation = dᵢ(PSᵢ) = ∑₃ⱼ=0 nᵢ,j(PSᵢ) sᵢ,j where PSᵢ is the perceptual signature associated with an image Iᵢ ∈ I.

The neural networks are three-layered feedforward networks, trained by backpropagation[18] with momentum. In our experiments, the input and output of each network are real numbers in [-1, 1]. The learning rate is arbitrarily set to 0.01 at the beginning of the learning. The maximum length of an epoch is 5000 iterations and the error threshold is set to 0.001. These values have been determined experimentally. Unlike typical applications where learning sets are constructed offline, in our implementation, continuously evolving training sets (a by-product of the user feedback) are considered. Since the back-propagation algorithm usually performs poorly at such task (catastrophic loss of memory for example), an analysis of the structure of each module and the subsequent learning set is performed so as to:

1. detect inconsistencies, i.e. similar perceptual states associated with impression words of contradictory meaning. Such inconsistencies typically originate from: (a) inappropriate selection of the perceptual subset, and (b) the attribution by the user of the same image for two different impression words in two different sessions.

2. detect clusters in the set of characteristics of a perceptual subset. The presence of clusters can be interpreted as an index of the use of nuances for a same impression word (e.g. "cold" as lifeless and "cold" as a lack of warmth).

3. limit its size (usually responsible for degenerescence of the weights (in conjunction with over-learning). A selection of the samples is performed
that keep the most recent ones and delete redundant ones. In our experiments, the size of the training set was set to a maximum of 20 positive samples and 20 negative ones.

Given a set of images not in the training set, the saliency value of each \( nn_j \) is computed as the correlation between network activation and user evaluation. In effect, the saliency value acts as a weight parameter to the output (activation) of a neural network. When the saliency is \(-1\), the actual activation of the network doesn’t have to be computed. A confidence \( c_j \) of each network \( nn_j \) is computed on the basis of the training set consistency, i.e. a) autocorrelation in the positive training set, b) cross-correlation between positive and negative training sets and c) accuracy of the learning process. The \( c_j \) confidence values are used by the system to compute a confidence value for the translation result (self-supervision by the feedback processing unit).

4.3. Words relational network

Over the course of interaction, new Kansei impressions will be encountered and new Kansei/impression words will be automatically introduced, i.e. a new word-module is constructed. In our implementation, word modules are connected into a network such as depicted in the upper right of Figure 1. The network consider synonymy, antonym, inclusion, correlation (e.g. melancholic and sad can seemingly be correlated) and nuance types of relationship. Each relation is associated with a weight (in the current implementation, weights are of value unity). A "nuance" relation corresponds to a hierarchical relation, i.e. when a word has nuances, the module of the nuance-related words will automatically be used instead of the original word-module. Relations are created and updated following user feedback (such as described in the following section) and self-supervised analysis of a word module’s training set (as described above). The interest of such network is three-fold: (1) efficiently dealing with a continuously expanding lexicon; when creating nuance relations, the respective training samples are pruned and the word-modules specialize; (2) bootstrapping the learning of new words: the use of synonymic and antonymic relations act as a selective learning mechanism and (3) providing both user and K-Agent with views on the language formation (externalization of the internal processes).

4.4. Feedback processing unit

The role of this unit is two-fold:

- an interface to the user in the form of externalization. In daily life, such process can take various forms, from conversation, written texts, sketch and memos, to simply physical records of actions. As mentioned by Miyake [19], externalized records are useful because they serve as sharable and concretely manipulable objects for constructive collaboration. In our implementation, it provides views on the structure of each word module (i.e. the operational factors of the translation process), the structure of the image processing kernel (i.e. allowing the integration of focus of attention) and the structure of the relational network between word-modules. In other publications [7, 20], we describe an interactive visual environment specially developped around the K-agent to enable both user and computational agent to engage in the modeling process. It is endowed with turn-taking capabilities so that both user and K-agent are truly active in the dialog.

- self-supervision of the learning units in the word-modules \( d_i \) to compensate for continuous learning (and the iterative construction of the training set) and structural modifications (i.e. fine tuning of the SB-segmentation in the image processing kernel, fine tuning of the saliency value \( s_j \) of neural network \( nn_j \), reinforcement or penalization of the learning weights of a neural network \( nn_j \) resulting from user feedback. Although unavoidable because user feedback is not uniform over time (interference of other cognitive processes or external factors), inconsistencies in the training sets of the modules \( d_i \) are critical to the performance of the K-agent. Back-propagation trained neural networks are particularly affected by an inconsistent distribution of the training samples and it typically results in catastrophic forgetting [21] and poor generalization. A filtering of the training sets is along the following lines:

- set an upper limit for the number of positive (resp. negative) samples. In our implementation, this limit was set to 20 (a trade-off
between learning performance and computational load);

- exclude images of similar PS signature and opposite evaluation. Instead, those images are used by the system to trigger interactive sessions with the user [7];

- set an upper limit to the interdistance between samples whose PS signature results in a similar evaluation. By avoiding the formation of cluster in the sample, it improves generalization. When clusters cannot be avoided, a nuance relation is created in the word relational network and the training set is splitted (see section 4.3.);

- restrict the sets to most recent samples. An infinite expansion of the training set is avoided to prevent from a degenerescence of the learning weights. This issue could also be partially addressed by using such technique as "weight-decay" [22] of the learning weights.

5. EVALUATION OF THE K-AGENT

In order to evaluate the feasibility and effectiveness of the concept of K-Agent, we selected a real-world application which requires the computer system to handle requests involving user Kansei. Nowadays, many websites (e.g. [23, 24]) have been set up for users to send virtual postcards to friends/relatives on various occasions such as birthday, dates, national holidays, etc. Typically, collections of images (a virtual postcard will be an image, a layout, a song, and a text entered by the user) are set for each category of card. It's quite unflexible and limited. It takes into account the sensibility (Kansei) of the user in so far as the user finds an item of his/her liking by browsing systematically the collection. When unsuccessful, the user will either have to switch onto another site or find a best compromise. On the other hand, many search-engines are available on the Internet for retrieving images, e.g. [25, 26, 27].

5.1. Protocol

The following scenario was considered: users were asked to perform that task using (1) their usual WEB-based search engine, i.e. querying the search engine using the objective keyword and then manually filtering against the subjective keyword; and (2) our own application (the implementation details of which are given in [20]) with the K-Agent filtering the results of the search engine. Users had been asked to train their own user model beforehand.

5.2. Experimental results

Figure 4 shows a graphical representation of the retrieving process, comparing the performance of the user manually browsing the results of AltaVista [26] and the result of our system. The horizontal axis denotes the number of images retrieved, while the vertical axis indicates the number of image browsed. The graph thus represents the "recall" for the two cases. With the impression word "cold", while the K-Agent still outperforms AltaVista, the difference in performance is not as significant as for the other impression words. The better performance of AltaVista essentially originates from the large presence of images conveying the impression "cold" in its collection. AltaVista naturally doesn't order its images with respect to the impression conveyed therefore its particular performance for the impression-word "cold" is marginal while the K-Agent performs consistently on all impression-words and the time required for the user to retrieve images is reduced very significantly. Because of the K-Agent's good performance, users were more inclined to engage in a search on a larger set of images, than when they had to do it manually. For example, a raw query of AltaVista on a set of keywords such as "Phuket", "Mountains" or "Airplanes" yielded thousands of images. In this experiment, we evaluated that in average a user had to browse up to 8 images before finding a satisfactory one. Typically, for the user to find 2 suitable images, s/he had to browse through 35 images (4 pages) in AltaVista while only 4 images were needed with the K-Agent based system.

6. FUTURE DIRECTIONS

Although we have discussed a general formalism for the translation between various modalities of impression expression, the modules translating from image to sound or image to movie, etc. must be specified. While the generalization of the verbalizing transformation is
straightforward, given a suitable specification of perceptual pre-processing, the generalization of translation between two media languages different from verbal one is more complicate due to the fact that media such as music and image have more complex structure than words. In particular, considering music for example, will involve a temporal dimension that has not been addressed here. In addition, our implementation is not necessarily optimal (e.g. back-propagation trained neural networks, restricted selection of visual processes) but is intended as an experimental platform for investigating the feasibility of the concept.

Another important extension is group modeling and indirect modeling of a population of user (i.e. the adaptation of an individual model to a population). Such capability will be essential in application areas such as fashion, product design, games, advertisement etc. where the computer system is targeted towards a population of user rather than a single individual. In entertainment, the system will be expected to quickly adapt to a new player by categorizing at run-time some profile in order to bootstrap the learning phase. Indirect modeling is more delicate. For example, consider the designer who must extrapolate on the translation language between the product characteristics and the targeted population of customers, while making abstraction of its own sensibilities. Group models will require not only the understanding of how impressions relate to the low-level features of the media but also the understanding of the relation between impressions and user profile (i.e. the parameters that unify a group of individuals). The criteria to extract similarity by user profile will differ according to the impression. For example, "romantic" will possibly relate to gender while "sadness" will relate to culture (e.g. difference of the meaning of the color black in chinese and european cultures). How to formalize such relations is still an open issue.

REFERENCES


15. A.H. Munsell; A Color Notation; Boston (1905).


