Expert Systems with Applications 36 (2009) 4929–4934

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Design facial appearance for roles in video games

Shang Hwa Hsu *, Ching-Han Kao, Muh-Cherng Wu

Department of Industrial Engineering and Management, National Chiao Tung University, 1001 Ta Hsueh Road, HsinChu 30010, Taiwan, ROC

ARTICLE INFO

Keywords:
Video game
Role
Fuzzy AHP
BP neural network

ABSTRACT

Roles in video games often serve as avatars of players. Different game players may have their particular preferences on a role's facial appearance. It would be desirable to allow players to customize the design of roles. This paper presents two methods for recommending a roles' facial appearance for a particular game player and illustrates the two methods by using heroic roles as an example. The two recommendation methods are designated as the *text-input* and the *picture-input* approaches. The text-input approach requests the game player to carry out pairwise comparisons for determining the relative weights of 16 personality traits of heroes. The recommendation mechanism for the text-input approach is based on the fuzzy AHP (analytic hierarchy process). Whereas the picture-input approach requests the game player to view a sample set of pictures and rate his/her preferences on each picture. The recommendation mechanism for the picture-input approach is based on the BP (back-propagation) neural network. Experiments indicated that the text-input approach is more effective in terms of recommending an appropriate facial appearance, yet at the expense of needing more user time.

1. Introduction

In playing a video game (called a game hereafter), a player expresses his or her intentions by manipulating the actions of a *role*. A game role is a character, which serves as the player's avatar or competitor. Roles to a game are as important as actors to a film. Casting appropriate actors can lead to the success of a film. In the same way, designing appropriate roles is very important to the success of a video game.

Previous researchers have published numerous studies on the role design of video games. Most of them attempted to create a life-like role by emulating human behaviors—such as dialogue (Brusk & Eladhari, 2006; Gustafson, Boye, Fredriksson, Johanneson, & Königsmann, 2005; Jan & Traum, 2005), intelligence (Frasca, 2001; Lair & Duchi, 2000; Vala, Paiva, & Prada, 2004), motor-skill actions (Blumberg & Galyean, 1997), and emotional expressions (Rizzo, Neumann, Enciso, Fidaleo, & Noh, 2001; Wallraven, Breidt, Cunningham, & Bülthoff, 2005).

Of these studies, those which analyze emotional expressions attempt to automatically create a facial expression to represent a character's mood (e.g. happy, angry, sad, and disgusted). Such facial expression studies have an implicit objective—*mood manifestation* (Bartlett, Hager, Ekman, & Sejnowski, 1999; Ekman, 1993; Zhang & Ji, 2005). That is, a computer-generated facial expression should model a particular mood (e.g. sad) that is easily recognizable by humans.

In a film, proper recognition of an actors' mood by interpreting their body languages is never enough. To produce a popular film, the attractive appearance of actors is often much more important. Likewise, the appearance of a game role may influence player involvement in the game. An attractive appearance may induce players to have affectionate feelings for the avatar, and in turn, make game play more fun (Hsu, Lee, & Wu, 2005). Even though the *facial appearance* of a role is very important to game design, this topic has rarely been investigated.

This study proposes two methods for automatically recommending attractive facial appearances of *heroic roles* to game players—in a *customization* manner. The observations in this study indicate that different game players have different preferences for the appearance of a hero. According to further interviews, this difference is due to the fact that players have different preferences for a hero's personality traits. This implies that *personality traits* may be manifested by *facial appearances.*

Based on this implication, this study proposes a research framework to design a hero's facial appearance as favored by a particular game player (Fig. 1). This research framework involves two phases: *creation* and *application*. The creation phase develops a database that relates facial appearance to personality traits, which involves two steps. Firstly, multimedia software is used to create samples of heroes' *facial appearances*. Secondly, game players evaluate the *personality traits* for each *facial appearance*. In this phase, a vector comprising numeric data can model both a facial appearance and its personality traits. The created database is called a *face-trait* database.

The application phase implements the customization design of the heroes' facial appearance. That is, for any given game player,

* Corresponding author. Tel.: +886 3 5726731; fax: +886 3 5722392.
E-mail address: shhsu@cc.nctu.edu.tw (S.H. Hsu).

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.05.049

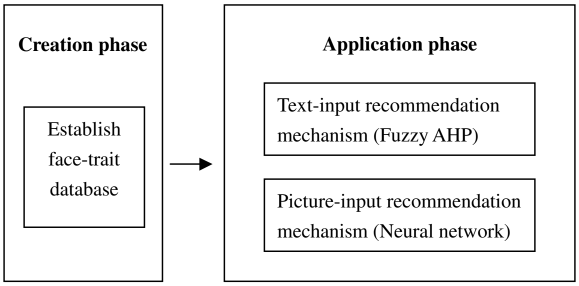

Fig. 1. Research framework.

a highly favored heroes' facial appearance can be quickly recommended from the face-trait database. Two recommendation mechanisms can be used to retrieve these highly favored facial appearance profiles.

One mechanism is a *text-input approach*, which requires the game player to determine each personality trait's relative weight. This mechanism uses the fuzzy AHP technique (Laarhoven & Pedrycz, 1983; Saaty, 1980; Shamsuzzaman, 2000; Zadeh, 1975). The other mechanism is a *picture-input* approach, which requires the game player to evaluate their degree of preferences for a sample set of pictures (facial appearances). This mechanism is implemented using the back-propagation (BP) neural network technique.

The remainder of this paper is organized as follows. Section 2 describes how the face-trait database is created. Section 3 first introduces the fuzzy AHP technique, and then presents the text-input recommendation mechanism. Section 4 describes the picture-input recommendation mechanism. Section 5 presents the experiments and results for justifying the effectiveness of the two recommendation mechanisms. The last section contains concluding remarks.

2. Face-trait database

A prototype *face-trait* database has been developed. In developing the database, heroes' *facial appearances* were created by commercially available multimedia software (Live Studio Head Tool V2.6). We use 16 *personality traits* of heroes to characterize each facial appearance created. A hero's *facial appearance* could then be encoded by a vector consisting of 16 elements, which provides a key for the recommendation facial appearances.

2.1. Facial appearance creation

The multimedia software used to create a facial appearance is based on a *feature-based* approach. That is, the configuration of a face is composed of several features. A feature may denote a part of a face such as *eyes*, *lips*, and *nose* or denote a face's characteristic such as *skin color* and *texture*. For each feature, there are many *options* for selection. Various combinations of feature options result in different facial appearances.

For features associated with geometric shapes, their *feature options* are created by the *parametric-geometry* approach (Myung & Han, 2001; Verroust, Schonek, & Roller, 1992). That is, changing some of its geometric parameters can vary the shape of an object. Considering an object with rectangular shape as an example, we could take the *length-to-width* ratio as one parameter and vary shape of the object by changing the parameter value. Likewise, a facial feature such as eyes can be modeled by a parametric-geometry shape with more than one parameter; for example, the curvature of upper eye lid and that of lower eye lid. By varying these two parameters, we could have many different shapes for modeling eyes.

Example features associated with geometric shapes include lips, noses, eyes, eyebrows, and facial outline. These shapes are commonly controlled by a set of parameters, and each parameter value

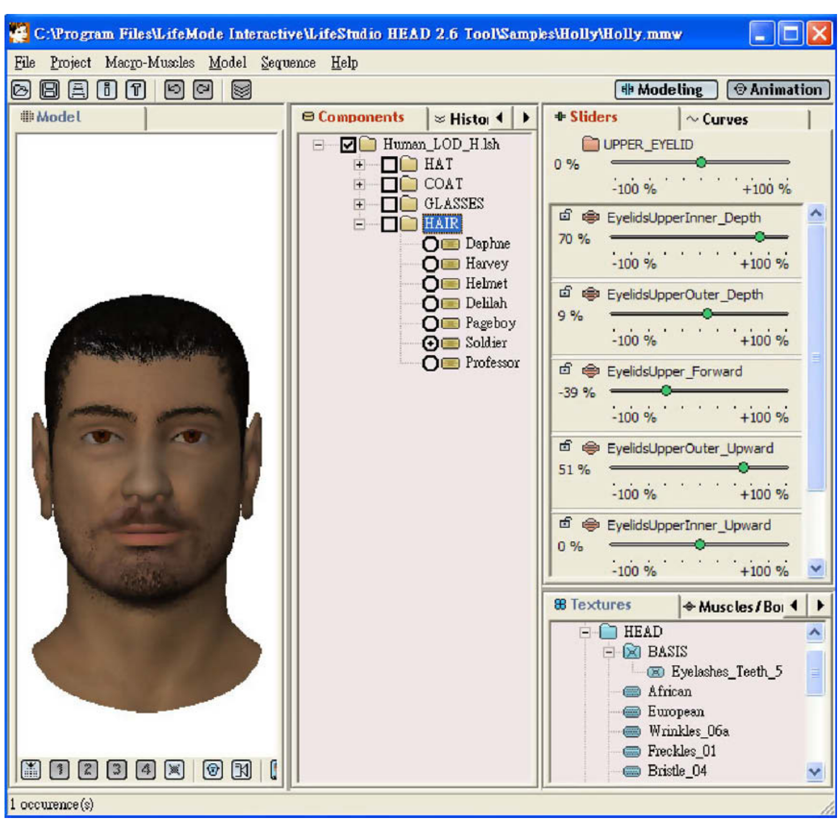

Fig. 2. The facial appearance creation software Live Studio Head Tool v.2.6.

is a real number in a predefined range such as [0, 1]. The input of such a real-number value is through a scale-bar input device (rightmost part of Fig. 2).

Due to the completeness property of real numbers, there are theoretically an infinite number of options available for features associated with geometric types. By contrast, some other features provide only a finite number of feature options; for example—type of hair and type of moustache. The input of such a nominal-type feature option is through the clicking of a menu box (middle part of Fig. 2).

The multimedia software we used provides 12 features. In this research, only five *shape-oriented* features are chosen as variables in designing a facial appearance for the prototype database. These five features are *facial outline*, *eye brows*, *eyes*, *lip*, and *nose*, which are selected because most prior research has concluded their importance on facial expression and recognition (Adolphs, 2002; Brunelli & Poggio, 1993; Sadrô, Jarudi, & Sinha, 2003).

The prototype database which is essentially expandable, at its present state, includes 243 different *facial appearances*. That is, each of the five facial features has only three options for selection, which leads to $3^5 = 243$ different *facial appearances*. Three instances of the facial appearances are shown in Fig. 3.

2.2. Personality traits evaluation

We use 16 *personality traits* to characterize each facial appearance. These personality traits are a hero's characters reported in a prior research by Hsu, Kao, and Wu (2007). According to their research, by the technique of factor analysis (principal components factoring) (McDonald, 1985), these 16 personality traits can be categorized into three groups—*bravery*, *visionary*, and *moral* as shown in Fig. 4.

In the characterization of a facial appearance, a five-point scale evaluates each personality trait. The higher the value, the higher degree does the facial appearance reveal—in terms of the personality trait. In the characterization process, we asked 112 subjects (61males, 51 females at the age of 17–25) to evaluate the 16 personality traits for each of 243 facial appearances. All these subjects are all game player, who are either senior high school or college students.

Fig. 3. Example of facial appearances.

Results of the characterization process yield 243 trait vectors, each of which represents a particular facial appearance. The value of each element, ranging from 1 to 5, is the mean score reported by all the subjects. Let $X = [x_1, \ldots, x_{16}]$ denote a trait vector so obtained, which is further transformed into a normalized vector, denoted by $Y = [y_1, \ldots, y_{16}]$ where $y_i = \frac{x_i}{\sum_{k=1}^{16} x_k}$ and $\sum_{i=1}^{16} y_i = 1$.

3. Text-input recommendation mechanism

To create a hero face favored by a particular game player, it is important to know how the player weights each personality trait. The procedure for determining the weights of personality traits is by applying the fuzzy AHP technique (Saaty, 1990). The technique, widely applied in various areas (Durán & Aguilo, 2007; Kim & Yoon, 1992; Mamaghani, 2002; Muralidar & Santhanam, 1990; Wu, Lo, & Hsu, 2007) is briefly stated below.

Firstly, the 16 personality traits are hierarchically clustered as shown in Fig. 4. This hierarchy indicates that these 16 personality traits, based on a prior research (Hsu et al., 2007), can be categorized into three groups—*bravery*, *visionary*, and *moral*, which are called *group-level traits*. Each personality traits within a group is called a *member-level trait*.

Secondly, the relative weights for *group-level* traits and that for *member-level* traits in each group have to be determined. We therefore have four *weight-determination* problems, one for group-level and three for member-level traits. To each weight-determination problem, we asked the game player to carry out a pairwise comparison experiment, and use a fuzzy AHP algorithm to process the experiment data for determining the relative weights.

Thirdly, we use the relative weights to compute a *recommendation-priority* value to retrieve a hero face from the face-trait database for the game player.

3.1. Pairwise comparison experiment

The pairwise comparison experiment is explained by using the weight-determination of group-level traits as an example. Of the three group-level traits, any two (a pair) is chosen for comparison. Assume the pair (*bravery, visionary*) is chosen. Then, the player is asked to answer the following question: "Consider a person to be recognized as a hero. Of the two groups of personality traits, which one is more important? Try to compare the degree of importance between them."

Now suppose the player answers *bravery* is more important than *visionary*. Then, five linguistic variables listed in Table 1 are provided for the player to express his/her opinion. These linguistic variables range from "absolutely important" to "equally important" on a five level scales, where between any two consecutive

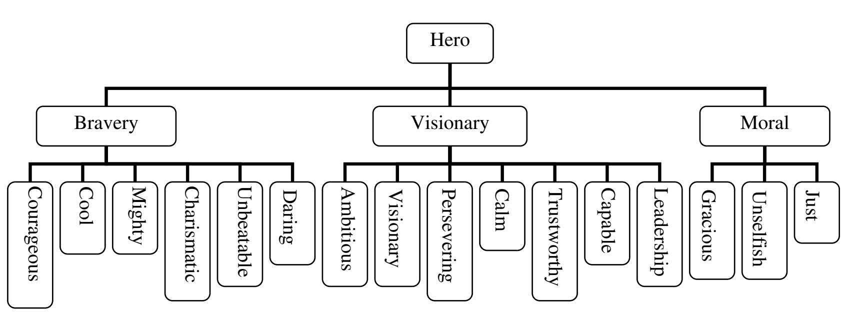

Fig. 4. Three levels fuzzy AHP hierarchy model.

Table 1
Linguistic variables used in the fuzzy AHP

Fuzzy number	Linguistic variables
$\tilde{1} = (1,1,1)$	Equally important
$\tilde{3} = (2,3,4)$	Weakly important
$\tilde{5} = (4,5,6)$	Essentially important
$\tilde{7} = (6,7,8)$	Very strongly important
$\tilde{9} = (8,9,10)$	Absolutely important
$\tilde{2} = (1,2,3); \tilde{4} = (3,4,5); \tilde{6} = (5,6,7); \tilde{8} = (7,8,9)$	Intermediate values between two adjacent judgments

scales an intermediate scale is additionally defined so that nine scales are finally created. As shown in Table 1, each linguistic variable is represented by a triangular fuzzy number—for example, $\tilde{7} = (6,7,8)$ denotes "very strongly important". The adoption of linguistic variables is to resolve the vagueness occurred in human judgment (Zadeh, 1975).

Based on the pairwise comparison experiment, a $n \times n$ matrix $\tilde{A} = [\tilde{a}_{ij}]$ could be obtained, where n denotes the number of traits to be compared,

$$\tilde{A} = \begin{bmatrix} 1 & \tilde{a}_{12} & \cdots & \tilde{a}_{1n} \\ \tilde{a}_{21} & 1 & \cdots & \tilde{a}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{a}_{n1} & \tilde{a}_{n2} & \cdots & 1 \end{bmatrix}$$

Notice that $\tilde{a}_{ij} = 1/\tilde{a}_{ji}$, which ensures that the comparison for each pair (i,j) is consistent; and $\tilde{a}_{ii} = 1$, *which denotes that a self-comparison is always "equally important" and is not needed.*

3.2. Fuzzy AHP algorithm

Two procedures are used to obtain the relative weights for *group-level* and *member-level* traits. The first one *Compute_Relative_Weight* is intended to compute the relative weights, and the second one *Validity_Check_for_Relative_Weights* is developed for checking the validity of the obtained relative weights. These two procedures are respectively described below, where the definitions of arithmetic operators (i.e., $\otimes$, $\oplus$, and *Defuzzy*) are introduced in the Appendix.

3.2.1. Procedure Compute_Relative_Weight

Step 1: Calculate $\tilde{W}_i$, the fuzzy weight for each row i in $\tilde{A}$ (Buckley, 1985)

$$\tilde{Z}_i = (\tilde{a}_i 1 \otimes \tilde{a}_i 2 \otimes \cdots \otimes \tilde{a}_{in})^{\frac{1}{n}}, \quad \forall i = 1,2,\ldots,n$$
$$\tilde{W}_i = \tilde{Z}_i \otimes (\tilde{Z}_1 \oplus \tilde{Z}_2 \oplus \cdots \oplus \tilde{Z}_n)^{-1}, \quad \forall i = 1,\ldots,n$$

Step 2: Defuzzication of $\tilde{W}_i$ and $\tilde{A}$ (Teng & Tzeng, 1993)

$$\hat{W}_i = Defuzzy(\tilde{W}_i)$$
$$a_{ij} = Defuzzy(\tilde{a}_{ij}), \text{ where} A = [a_{ij}] \text{ and } \tilde{A} = [\tilde{a}_{ij}]$$

Step 3: Compute relative weights W_{i}

$$W_i = \frac{\hat{W}_i}{\sum_{i=1}^{n} \hat{W}_i}$$

3.2.2. Procedure Validity_Check_for_Relative_Weights

Step 1: Compute W_i^*

$$\begin{bmatrix} W_1^* \\ W_2^* \\ \vdots \\ W_n^* \end{bmatrix} = A \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \end{bmatrix}$$

Step 2: Compute $\lambda_{\max}$, the maximum eigenvalue

$$\lambda_{\max} = \frac{1}{n}\left[\left(\frac{W_1^*}{W_1}\right) + \left(\frac{W_2^*}{W_2}\right) + \cdots + \left(\frac{W_n^*}{W_n}\right)\right]$$

Step 3: Compute CI, the consistency index

$$\mathrm{CI} = \frac{\lambda_{\max} - n}{n - 1}$$

Step 4: Compute CR

CR = CI/RI the values of RI are shown in Table 2 (Saaty, 1980).

Step 5: Consistency check

If $\mathrm{CR} \leqslant 0.1$ (pairwise comparison data $\tilde{A}$ is reasonably consistent)

Output the resulting relative weights W_i *($1 \leqslant i \leqslant n$)*

Else (CR > 0.1 pairwise comparison data $\tilde{A}$ are inconsistent)

Repeat the pairwise comparison experiment.

Endif

3.3. Recommending mechanism

To recommend a hero face favored by a particular game player, we first compute a *recommendation-priority* value for each facial appearance in the face-trait database and recommend the one with the highest recommendation-priority value.

Define $S = \{(Y_i, F_i) | 1 \leqslant i \leqslant n\}$ as the face-trait database developed for facial design, where F_i denotes ith facial appearance, and $Y_i = [y_{ij}]$, $1 \leqslant j \leqslant 16$ denotes the *personality-trait* vector of F_i Notice that F_i is a 2D picture while Y_i is a vector with 16 numeric elements, where y_{ij} denotes the degree of jth personality traits that picture F_i manifest to a "*common people*". Let $W = [w_j]$, $1 \leqslant j \leqslant 16$ represent the preferences of a particular game player on each of the 16 personality traits, where w_j denotes the relative weight of jth personality trait, perceived by the game player—an "*individual people*" rather than a "*common people*".

The procedure for recommending facial appearances most favorable to a particular game player proceeds as follows.

Step 1: Compute the recommendation-priority value

$$p_i = \sum_{j=1}^{16} w_j \cdot y_{ij}; \quad 1 \leqslant i \leqslant n$$

Step 2: Output the recommended one

$$i^* = \arg\max(p_i)$$

4. Picture-input recommendation mechanism

The BP neural network technique has been widely used as a predictor (Dutta & Shekhar, 1988; Liau & Chen, 2005; O'Leary, 1998; Salchenberger, Cinar, & Lash, 1992; Tam & Kiang, 1992). Given a sample set of input/output data obtained from a real-world system, we can use the technique to establish a BP network that serves as an input/output mapping mechanism—a three-layer architecture as shown in Fig. 5. The established BP network can be used to predict the output for a new input data set. Details of the BP neural network technique can be referred to Wasserman (1989) and Hertz, Krogh, and Palmer (1991).

The picture-input recommendation mechanism is to establish a BP network for a particular game player. For such a BP network, its

Table 2
Values of RI

n	1	2	3	4	5	6	7	8	9	10	11
RI	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51

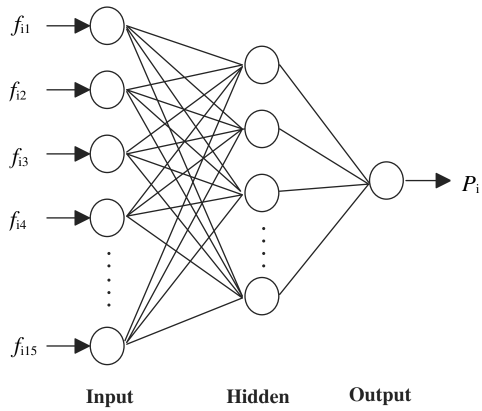

Fig. 5. Architecture of the neural network.

input is a facial appearance denoted by $F_i = [f_{ik}]$, $1 \leqslant i \leqslant 243$, $1 \leqslant k \leqslant 15$, where f_{ik} (a binary number) denotes the existence of the kth *feature option* in the ith facial appearance. As stated, a facial appearance has five features, each of which has three options for selection. This leads to 15 $(5 * 3)$ types of feature options. A facial appearance can then be modeled by a vector with 15 binary elements, in which 1 denotes a feature option exists and 0 denotes its nonexistence. The output of such a BP network is denoted by P_i, which denotes the game player's preference on the ith facial appearance.

The use of the BP neural network technique involves two phases: *training* and *predicting* phases. In the training phase, we attempt to establish a BP network for the game player. Out of the 243 facial appearances, we randomly sample 81 ones and use them to build up a BP network. The algorithms of the training process can be referred to Grossberg (1974) and Rumelhart, Hinton, and Williams (1986). In the predicting phase, we attempt to use the established BP network to predict the player's preferences on the other facial appearances not considered in training phase. That is, the BP network will predict the player's preference on the remaining 162 facial appearances.

5. Experiments

An experiment is carried out to compare the performance of the *text-input* and the *picture-input* approaches. Twenty game players (10 males, 10 females at the age of 17–25) are invited as experiment subjects, and 243 facial appearances are created in the face-trait database. The experiment proceeds as follows.

Firstly, we carried out the text-input approach in which each subject i is requested to perform a pairwise comparison for determining his/her relative weights on the 16 personality traits. The system will output a picture (say, X_i^*) with the highest recommendation-priority value.

Secondly, we carried out the picture-input approach in which each subject i is requested to view 81 pictures and give his/her preference on each picture. The input/output data of the 81 pictures are used to establish a BP network. Then, all the 243 pictures are fed into the BP network, and the one (say, Y_i^*) with the highest preference value will be recommended.

Finally, each subject is requested to view the remaining 162 pictures and give his/her preference on each one. Out of the 243 pictures, the one with the highest preference (say, Z_i^*) is selected.

Define $r(P_i^*)$ as the preference of picture P evaluated by subject i. The effectiveness of the text-input approach is measured by the distribution of the metric: $x_i = r(Z_i^*) - r(X_i^*)$, and that for the picture-input approach is measured by the distribution of $y_i = r(Z_i^*) - r(Y_i^*)$.

The distributions of x_i and y_i are shown in Table 3. The table indicates that the text-input approach is superior to the picture-input approach. In the text-input approach, 85% game players (17 out of the 20 subjects) will get their most favor picture, while only 35% can get so in the picture-input approach. However, the text-input approach needs more user time. The average time for performing a pairwise comparison is about 26 min. and that for evaluating 81 pictures is about 2.6 min.

The experiment results lead to the following implication. For a video game equipped with relatively few numbers of pictures, we would suggest an exhaustive display of pictures to a game player. In contrast, for a video game equipped with a great amount of pictures, we would suggest the use of the text-input approach.

6. Conclusions

Playing video games through manipulating a role is quite common. Role design therefore has been a significant research issue. Most prior research attempted to create a life-like role by develop software for emulating human's capabilities, such as dialogues, intelligence, motor-skill, and emotional expressions. This research is unique in providing a customized facial appearance for each game player. A video game with such a customization function would become more popular.

Two methods for providing such a customization function have been developed. One is called the text-input approach whose mechanism is based on the fuzzy AHP technique. The other is called the picture-input approach whose mechanism is based on the BP neural network technique. The text-input approach requires a game player to perform a pairwise comparison on 16 personality traits. The picture-input approach requires a game player to view and evaluate a sample set of pictures.

Experiment results indicated that the text-input approach is superior to the picture-input approach, in terms of recommending an appropriate picture to a game player; however, at the price of needing more user input time.

Appendix. Arithmetical operators for fuzzy numbers

The fuzzy arithmetic operators used in this research are defined below (Laarhoven & Pedrycz, 1983; Zadeh, 1975), by referring two fuzzy numbers $\tilde{a}_1 = (l_1, m_1, r_1)$ and $\tilde{a}_2 = (l_1, m_1, r_1)$.

(1) Addition operator: $\oplus \tilde{a}_1 \oplus \tilde{a}_2 = (l_1 + l_2, m_1 + m_2, r_1 + r_2)$
(2) Multiplication operator: $\otimes \tilde{a}_1 \otimes \tilde{a}_2 = (l_1 \times l_2, m_1 \times m_2, r_1 \times r_2)$
(3) Defuzzication operator (Teng & Tzeng, 1993) $Defuzzy(\tilde{a}_1) = |(r_1 - l_1) + (m_1 - l_1)|/3 + l_1$

Table 3
Comparing effectiveness

Subjects	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Text-input x_i	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0
Picture-input y_i	0	0	3	0	1	2	0	2	0	1	0	1	1	2	0	2	1	1	3	1

References

Adolphs, R. (2002). Recognizing emotions from facial expressions: Psychological and neurological mechanisms. *Behavioral and Cognitive Neuroscience Reviews 1, 1*, 21–61.

Bartlett, M. S., Hager, J. C., Ekman, P., & Sejnowski, T. J. (1999). Measuring facial expressions by computer image analysis. *Psychophysiology, 36*, 253–263.

Blumberg, B., & Galyean, T. (1997). Multi-level control for animated autonomous agents: Do the right thing... oh, not that.... In: R. Trappl & P. Petta (Eds.), *Creating personalities for synthetic actors*. Berlin: Springer-Verlag. (pp. 74–82).

Brunelli, R., & Poggio, T. (1993). Face recognition features versus templates. *IEEE Transactions on PAMI, 15*(10), 1042–1052.

Brusk, J., & Eladhari, M. (2006). Playing the character. *RPG seminar, Tampere, March* (pp. 30–31).

Buckley, J. J. (1985). Ranking alternatives using fuzzy numbers. *Fuzzy Sets and Systems, 15*(1), 21–31.

Durán, O., & Aguilo, J. (2007). Computer-aided machine-tool selection based on a Fuzzy-AHP approach. *Expert Systems with Applications*. doi:10.1016/j.eswa.2007.01.046.

Dutta, S., & Shekhar, S. (1988). Bond rating: A non-conservative application of neural networks. In *Proceedings of the IEEE international conference on neural networks* (pp. 443–450).

Ekman, P. (1993). Facial expressions and emotion. *American Psychologist, 48*, 384–392.

Frasca, G. (2001). Rethinking agency and immersion: Videogames as a means of consciousness-raisin (essay). *SIGGRAPH 2001 N-Space art gallery*.

Grossberg, S. (1974). Classical and instrumental learning by neural networks. *Progress in theoretical biology* (Vol. 3, pp. 51–141). New York, NY: Academic Press.

Gustafson, J., Boye, J., Fredriksson, M., Johanneson, L., & Königsmann, J. (2005). Providing computer game characters with conversational abilities. In *Proceedings of intelligent virtual agent (IVA05), Kos, Greece*.

Hertz, J., Krogh, A., & Palmer, R. G. (1991). *Introduction to the theory of neural computation*. Redwood City, CA:: Addison-Wesley.

Hsu, S. H., Kao, C. H., & Wu, M. C. (2007). Factors influencing player preferences for heroic roles in role-playing games. *CyberPsychology & Behavior, 10*(2), 293–295.

Hsu, S. H., Lee, F. L., & Wu, M. C. (2005). Designing action games for appealing to buyers. *CyberPsychology & Behavior, 8*(6), 585–591.

Jan, D., Traum, D. R. (2005). Dialog simulation for background characters. In *IVA* (pp. 65–74).

Kim, C. S., & Yoon, Y. (1992). Selection of a good expert system shell for instructional purposes in business. *Information & Management, 23*(5), 249–262.

Laarhoven, P. J. M., & Pedrycz, W. (1983). A fuzzy extension of Saaty's priority theory. *Fuzzy Sets and Systems, 11*(3), 229–241.

Lair, J. E., & Duchi, J. C. (2000). Creating human-like synthetic characters with multiple skill levels: A case study using the Soar Quakebot. *AAAI 2000 fall symposium on simulating human agent, technical report FS-00-03* (pp. 75–79). AAAI Press.

Liau, L. C. K., & Chen, B. S. C. (2005). Process optimization of gold stud bump manufacturing using artificial neural networks. *Expert Systems with Applications, 29*, 264–271.

Mamaghani, F. (2002). Evaluation and selection of an antivirus and content filtering software. *Information Management & Computer Security, 10*(1), 28–32.

McDonald, R. (1985). *Factor analysis and related techniques*. Hillsdale, NJ: Lawrence Erlbaum.

Muralidar, K., & Santhanam, R. (1990). Using the analytic hierarchy process for information system project selection. *Information & Management, 18*(1), 87–95.

Myung, S., & Han, S. (2001). Knowledge-base parametric design of mechanical products based on configuration design method. *Expert Systems with Applications, 21*, 99–107.

O'Leary, D. E. (1998). Using neural networks to predict corporate failure. *International Journal of Intelligent Systems in Accounting, Finance and Management, 7*, 187–197.

Rizzo, A. A., Neumann, U., Enciso, R., Fidaleo, D., & Noh, J. Y. (2001). Performance-driven facial animation: Basic research on human judgments of emotional state in facial avatars. *CyberPsychology & Behavior, 4*(4), 471–487.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal representation by error propagation. *Parallel and Distributed Processing, 1*, 318–362.

Saaty, T. L. (1980). *The analytic hierarchy process: Planning, priority setting, resource allocation*. New York: McGraw-Hill (p. 20).

Saaty, T. L. (1990). How to make a decision: The analytic hierarchy process. *European Journal of Operational Research, 48*(1), 9–26.

Sadrô, J., Jarudi, I., & Sinha, P. (2003). The role of eyebrows in face recognition. *Perception, 32*, 285–293.

Salchenberger, L. M., Cinar, E. M., & Lash, N. A. (1992). Neural networks: A new tool for predicting thrift failures. *Decision Sciences, 23*(4), 899–916.

Shamsuzzaman, M. (2000). Selection of a FMS based on fuzzy set theory and AHP methods. ISE-Thesis, Asian Institute of Technology, Bangkok.

Tam, K. Y., & Kiang, M. Y. (1992). Managerial applications of neural networks: The case of bank failure predictions. *Management Science, 38*(7), 926–947.

Teng, J. Y., & Tzeng, G. H. (1993). Transportation investment project selection with fuzzy multi-objective. *Transportation Planning and Technology, 17*, 91–112.

Vala, M., Paiva, A., & Prada, R. (2004). From motion control to emotion influence: Controlling autonomous synthetic characters in a computer game. In *AAMAS'04, New York, USA* (pp. 1300–1301).

Verroust, A., Schonek, F., & Roller, D. (1992). Rule-oriented method for parameterized computer-aided design. *Computer Aided Design, 24*(10), 531–540.

Wallraven, C., Breidt, M., Cunningham, D. W., & Bülthoff, H. H. (2005). Psychophysical evaluation of animated facial expressions. In *Proceedings of the 2nd symposium on Applied perception in graphics and visualization* (pp. 26–28).

Wasserman, R. (1989). *Neural computing: Theory and practice*. New York, NY: Van Nostrand Reinhold.

Wu, M. C., Lo, Y. F., & Hsu, S. H. (2007). A fuzzy CBR technique for generating product ideas. *Expert Systems with Application, 34*, 530–540.

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences, Part 1, 8*, 199–249; *Part 2, 8*, pp. 301–357; *Part 3*, pp. 43–80.

Zhang, Y., & Ji, Q. (2005). Active and dynamic information fusion for facial expression understanding from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(5), 699–714.