

Form design of product image using grey relational analysis and neural network models

Hsin-Hsi Lai^a, Yang-Cheng Lin^a, Chung-Hsing Yeh^{b,*}

^a*Department of Industrial Design, National Cheng Kung University, Tainan, 701, Taiwan, ROC*

^b*School of Business Systems, Monash University, Clayton, Victoria, 3800, Australia*

Accepted 29 March 2004

Abstract

This paper presents a new approach to determining the best design combination of product form elements for matching a given product image represented by a word pair. A grey relational analysis (GRA) model is used to examine the relationship between product form elements and product image, thus identifying the most influential elements of product form for a given product image. A grey prediction (GP) model and a neural network (NN) model are used individually and in conjunction with the GRA model, in order to predict and suggest the best form design combination. An experimental study on the form design of mobile phones is conducted to evaluate the performance of these models. Based on expert surveys, the concept of Kansei Engineering is used to extract and evaluate the experimental samples, and a morphological analysis is used to extract form elements from these sample mobile phones. The evaluation result shows that all the NN-based models outperform the GP-based models, suggesting that the NN model should be used to help product designers determine the best combination of form elements for achieving a desirable product image. The GRA model can be incorporated into the NN model to help designers focus on the most influential elements in form design of mobile phones.

© 2004 Elsevier Ltd. All rights reserved.

Keywords: Product form; Product image; Neural networks; Kansei Engineering; Grey relational analysis; Grey prediction

1. Introduction

The image of a product plays an important role in consumers' preference and choice of the product [1]. Whether consumers choose a product depends largely on their perception of the product image [2]. Based on the relationship between the product form and the product image perceived

* Corresponding author. Tel.: +61-3-9905-5808; fax: +61-3-9905-5159.

E-mail address: chunghsing.yeh@infotech.monash.edu.au (C.-H. Yeh).

by consumers, design support models [1,3–5] and consumer-oriented technologies [6,7] have been developed to help designers design product form for a given product image. In particular, Kansei Engineering [8] has been developed as “translating technology of a consumer’s feeling (Kansei in Japanese) and image of a product into design elements”. It has been applied successfully in the product design field [9–12] to explore the relationship between the feeling (perception of the product image) of the consumers and the design elements of the product.

In this paper, we present a new approach for answering specific research questions in product design with respect to product form and product image, using the grey system [13] and neural network (NN) [14] techniques. These research questions include (a) how the product form elements affect a particular image of the product, (b) how the product form elements can be best combined to match a desirable product image, and (c) what technique should be used to help product designers determine the best combination of product form elements for a given product image. To illustrate how the approach can answer these questions, we conduct an experimental study on mobile phones for their popularity as a consumer product and their wide variety of product form elements.

In subsequent sections, we first describe how Kansei Engineering and morphological analysis can be used to extract representative samples and form elements of mobile phones as numerical data sets required for analysis. We then present the techniques used to analyze the experimental data sets for answering the research questions, and discuss the results of applying these techniques. Finally, we evaluate the performance of these techniques in order to determine the best model that can be used to help design form elements for matching a designated product image.

2. Extracting and evaluating experimental samples using Kansei Engineering

Kansei Engineering is a process of linking the consumer’s feeling (Kansei) of a product, represented by an image word pair, to the product design elements, using a survey or an experiment [8]. In this paper, we conduct an experimental study using the concept of Kansei Engineering to collect numerical data about the relationship between an image word pair and form elements of mobile phones. The subjects of the experiment consist of three groups of product design experts. The first group has eight males and seven females for extracting the representative samples of mobile phones. The second group has two males and three females for performing the morphological analysis to extract form elements of mobile phones. The third group has eight males and seven females for evaluating the product image of experimental samples, whose result is used as a basis for evaluating the performance of the models developed in this study.

2.1. *Extracting representative samples of mobile phones*

To facilitate the identification of commonly used form elements of mobile phones in the market, we need to first classify mobile phones based on their similarity degree. This classification result can then be used to extract sample mobile phones for identifying common form elements and for subsequent model building and testing. This procedure involves the following nine steps:

Step 1: Select 54 mobile phones of various makers and models, including 30 mobile phones used in our previous study [12,15] and 24 mobile phones entering the market during 2000–2001.

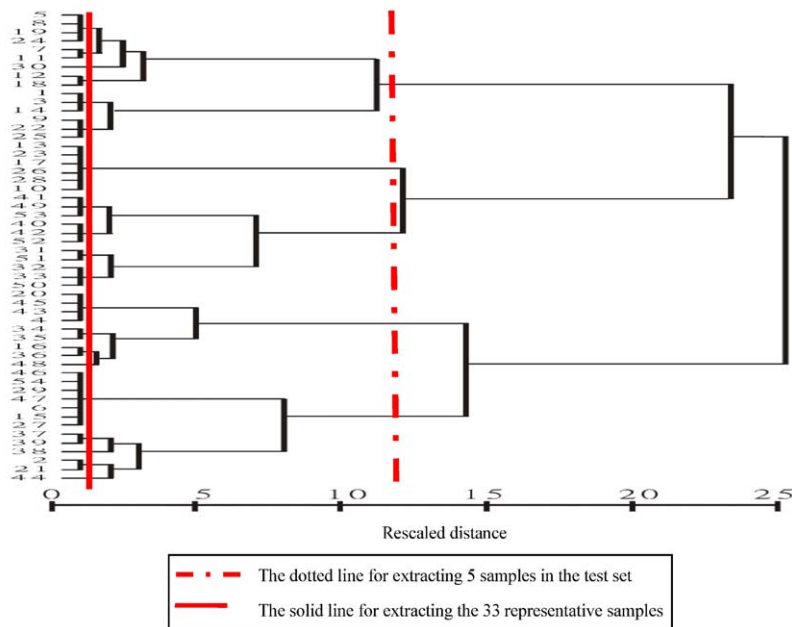


Fig. 1. Dendrogram of 54 mobile phones.

Step 2: Make 54 small paper cards according to the original size of each mobile phone.

Step 3: Separate 54 small paper cards into 7–12 groups by their similarity degree, based on the experimental result obtained from the 15 subjects of the first expert group using the Kawakida Jirou (K.J.) method [16]. The method was introduced by Kawakida Jirou in 1953 for classifying ideas, concepts, or objects into several groups. It has been successfully applied to a variety of classification problems.

Step 4: Build a similarity matrix from the separation result obtained at Step 3.

Step 5: Transform the similarity matrix into a dissimilarity matrix for the analysis at Step 6.

Step 6: Apply the multidimensional scaling (MDS) analysis to the dissimilarity matrix data. To determine the most appropriate dimensionality for the data, we examine five different dimensional spaces (ranging from two to six dimensions), which are commonly used in empirical studies.

Step 7: Choose six dimensions as a result of the MDS analysis with stress=0.10 and RSQ=0.83. In determining the dimensionality to use for a given set of data, a commonly used measure of fit in MDS is “stress”, which is the square root of a normalized residual sum of squares. A smaller stress value indicates a better fit. Another common measure of fit is the squared correlation (RSQ) value. The higher the RSQ value, the better the fit. The stress values for the two–six dimensions examined are 0.31, 0.22, 0.16, 0.13, and 0.10, respectively, and the RSQ values are 0.50, 0.62, 0.73, 0.78, and 0.83, respectively. Thus, the six-dimensional space is the most appropriate.

Step 8: Perform cluster analysis based on the MDS result. Fig. 1 shows the average linkage dendrogram (cluster tree diagram) of 54 mobile phones. The horizontal axis in Fig. 1 indicates the average squared Euclidean distances of the clusters agglomerated, which are rescaled to values between 0 and 25 for easy reference.

Step 9: Extract the 33 representative mobile phone samples, including 28 samples as the training set and five samples as the test set for training and testing the models to be built in the paper. The 33 samples are selected by drawing the solid line on the dendrogram in Fig. 1, resulting in 33 clusters being formed. This number is closest to the half of the total mobile phones investigated and is manageable by the survey and experimental processes. To use a relatively small set of samples as the test set, we choose a distance (as indicated by the dotted line in Fig. 1) near half of the largest distance and its resultant number of clusters (five in this case) is reasonable. The five mobile phones as the corresponding five cluster centers (i.e. nos. 14, 23, 29, 33, and 45) are thus used as the test set. The 21 unselected mobile phones will be used as an out-of-sample test set.

2.2. Morphological analysis

In this study, the product form of mobile phones is represented by both the outline shapes and the product elements. We use the morphological analysis to extract the form elements of the 33 mobile phone samples. The morphological analysis involves two steps. In the first step, the five subjects of the second expert group are asked to write down the influential form elements of the mobile phones individually according to their knowledge and experience. The survey results are grouped into two parts: form feature and form treatment. The form feature part includes the size and shape of outline components that make up the mobile phone, such as buttons, icons, the screen, or the body shell. The form treatment part indicates the relationship between the outline components, for example, the equidistance arrangement of the buttons or the size rate of the screen and the body shell. In the second step, the five experts form a focus group [17] to combine similar opinions of the survey results.

As a result of morphological analysis, Table 1 shows the nine form elements extracted from the 33 mobile phone samples, together with their associated types. Each form element has different types of its own, ranging from 2 to 4, as indicated by the type number 1, 2, 3 or 4 in Table 1.

2.3. Evaluating product image of experimental samples

To evaluate the degree to which the 33 mobile phone samples match a given product image, the 15 subjects of the third expert group are involved in the experimental evaluation of Kansei Engineering [7,8]. In Kansei Engineering, surveys or experiments are conducted to grasp the consumer's Kansei (psychological feeling and image) about a new product using the semantic differentials (SD) method [18]. Pairs of Kansei words are often used to describe the consumer's image of the product in terms of ergonomic and psychological estimation. With the identification of the design elements of the product, the relationship between the Kansei (image) words and the design elements can be established.

In our study, the image word pair used is simple–complex (S–C) and the design elements used are the form elements of mobile phones as identified in Table 1. The S–C word pair has the highest predictive consistency, as compared to other image word pairs such as Handsome–Rustic and Leisure–Formal, according to our previous studies [12,15]. To obtain the evaluation value for the S–C image of a mobile phone, a 7-point scale (1–7) of the SD method is used. As such, the 15 subjects are asked to assess the form (look) of mobile phones on a simplicity–complexity scale of 1–7, where 1 is most simple and 7 is most complex. The last four columns of Table 2 show

Table 1

Form elements extracted by morphological analysis on 33 mobile phone samples







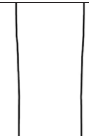






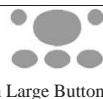
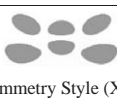

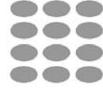










Elements	Type 1	Type 2	Type 3	Type 4
1. Top Shape (X ₁)	 Line (X ₁₁)	 Curve (X ₁₂)	 Arc (X ₁₃)	 Irregular (X ₁₄)
2. Body Shape (X ₂)	 Parallel Line (X ₂₁)	 Raised Curve (X ₂₂)	 Concave Curve (X ₂₃)	
3. Bottom Shape (X ₃)	 Line (X ₃₁)	 Curve (X ₃₂)	 Arc (X ₃₃)	
4. Length and Width Ratio of Body (X ₄)	 Wide Ratio 2:1 (X ₄₁)	 Middle Ratio 2.5:1 (X ₄₂)	 Slender Ratio 3:1 (X ₄₃)	
5. Buttons Style (X ₅)	 With Large Button (X ₅₁)	 Symmetry Style (X ₅₂)	 Other Style (X ₅₃)	
6. Number Buttons Arrangement (X ₆)	 Regular (X ₆₁)	 Irregular (X ₆₂)		
7. Screen Size (X ₇)	 TV Ratio 4:3 (X ₇₁)	 Movie Ratio 16:9 (X ₇₂)	 Other Ratio (X ₇₃)	
8. Screen Mask and Function Buttons (X ₈)	 Independence (X ₈₁)	 Function Buttons Dependence on Screen Mask (X ₈₂)	 Interdependence (X ₈₃)	
9. Outline Division Style (X ₉)	 Normal Division (X ₉₁)	 Rim Division (X ₉₂)	 Special Division (X ₉₃)	

Table 2

Kansei evaluation matrix for 33 selected mobile phones

Phone no.	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	S–C value			
										Average	Min	Max	Standard deviation
2	2	2	1	3	2	2	1	2	1	4.467	2	7	1.500
4	2	3	1	2	3	2	1	2	3	3.133	1	7	1.746
7	1	3	2	1	2	1	2	3	1	2.867	1	6	1.707
10	3	3	1	2	2	1	3	1	1	5.533	3	7	1.310
12	4	1	1	2	3	2	3	1	2	5.533	3	7	1.310
14*	1	2	1	2	2	1	1	3	1	6.400	5	7	0.712
16	2	2	1	2	2	1	2	2	3	6.867	5	7	0.499
18	4	1	2	3	2	2	1	2	3	6.800	5	7	0.542
19	2	2	1	1	1	1	2	1	2	4.600	2	7	1.781
22	2	3	1	2	2	1	1	1	1	6.200	4	7	1.108
23*	3	2	3	3	2	2	1	1	2	6.600	4	7	0.879
24	2	1	1	2	2	1	2	3	1	4.533	2	7	1.857
25	1	2	1	2	1	1	3	3	1	3.933	1	6	1.436
26	3	2	2	3	1	1	2	3	3	6.933	6	7	0.249
27	2	2	1	3	1	1	1	3	1	5.600	2	7	1.993
29*	1	1	1	3	1	1	1	1	1	2.733	1	7	1.652
30	1	1	1	2	3	1	2	2	3	4.333	1	7	2.022
31	2	2	1	2	1	1	2	3	3	2.533	1	5	1.360
33*	2	3	2	2	1	1	2	1	2	4.533	2	7	1.821
34	2	2	2	2	1	1	1	2	1	3.200	1	7	1.720
35	2	2	2	2	2	1	1	2	1	5.533	4	7	1.147
36	2	2	2	2	3	1	1	1	1	6.867	6	7	0.340
38	2	2	1	2	2	2	1	2	3	4.467	2	7	1.586
39	2	2	1	2	2	1	2	2	1	3.667	1	6	1.445
41	3	3	1	2	1	1	2	2	3	3.067	1	7	2.015
42	3	3	3	3	2	1	2	3	1	4.533	1	7	1.586
44	3	2	1	3	2	2	1	2	1	3.600	1	7	1.818
45*	1	2	2	2	1	2	1	2	3	4.600	2	7	1.993
46	2	2	2	3	1	1	1	1	1	4.933	2	7	1.731
48	1	3	1	3	1	1	1	1	1	5.333	3	7	1.535
50	2	3	3	1	1	1	1	3	3	3.733	1	7	2.112
51	2	3	3	2	1	2	2	3	2	3.533	1	7	1.746
52	2	1	3	2	2	1	2	2	2	3.733	1	7	1.843

the evaluation results of the 33 selected samples (out of 54), including 28 samples in the training set and five samples in the test set (asterisked), as identified in Fig. 1. For each selected mobile phone in Table 2, the first column shows the mobile phone number and columns 2–10 show the corresponding type number for each of its nine form elements, as given in Table 1. As shown in Table 2, mobile phone no. 31 has the “simplest” product image with an average S–C value of 2.533, while mobile phone no. 26 has the “most complex” product image with an average S–C value of 6.933.

Table 2 provides a numerical data source for applying the grey system and NN techniques to answer the three research questions identified in this study. To help present how these questions are dealt with, these techniques are described in the following section.

3. Grey system and NN techniques

The grey system theory [13] has been developed to examine the relationship among factors in an observable system where the information available is grey, meaning uncertain and incomplete (i.e. only part of the information is known). It has been successfully used in a wide range of fields, including some recent application results [19–23] highlighting its effective handling of incomplete known information for exploring unknown information. The system that can be built for answering specific research questions in product design with respect to product form and product image is grey in essence, as there is no way to identify all the product form elements that affect a particular product image perceived by consumers. To address the research questions in this study, we use two techniques of the grey system theory: grey relational analysis (GRA) and grey prediction (GP).

3.1. Grey relational analysis (GRA)

The GRA is used to determine the relationship (similarity) between two series of stochastic data in a grey system. One is the reference series ($x_0 \in X$), and the other is the m comparison series ($x_i \in X$, $i = 1, 2, \dots, m$), where $X = \{x_\sigma | \sigma = 0, 1, 2, \dots, n\}$ is a given grey relational element set. The grey relational degree between the two series at a certain time point t is represented by the grey relational coefficient $r(x_0(k), x_i(k))$, defined as

$$r(x_0(k), x_i(k)) = \frac{\min_i \min_k |x_0(k) - x_i(k)| + \xi \max_i \max_k |x_0(k) - x_i(k)|}{|x_0(k) - x_i(k)| + \xi \max_i \max_k |x_0(k) - x_i(k)|},$$

$$k = 1, 2, \dots, n; \quad i = 1, 2, \dots, m, \quad (1)$$

where $\xi \in [0, 1]$ is a distinguishing coefficient for controlling the resolution scale, usually being assigned the value of 0.5. The grey relational degree of each comparison series x_i ($i = 1, 2, \dots, m$) to reference series x_0 at all time points can be calculated by

$$r(x_0, x_i) = \frac{1}{n} \sum_{k=1}^n r(x_0(k), x_i(k)), \quad i = 1, 2, \dots, m. \quad (2)$$

If $r(x_0, x_i) > r(x_0, x_j)$, then the element x_i is closer to the reference element x_0 than the element x_j . The grey relational coefficient has the following properties [13]:

(1) Space norm interval:

$$0 < r(x_0, x_i) \leq 1, \quad \forall i,$$

$$r(x_0, x_i) = 1 \Leftrightarrow x_0 = x_i,$$

$$r(x_0, x_i) = 0 \Leftrightarrow x_0 \cap x_i \in \phi.$$

(2) *Space duality symmetric*:

$$r(x_0, x_i) = r(x_i, x_0) \Leftrightarrow X = \{x_0, x_i\}.$$

(3) *Space wholeness*:

$$r(x_i, x_j) \neq r(x_j, x_i) \Leftrightarrow x_i, x_j \in X = \{x_\sigma | \sigma = 0, 1, 2, \dots, n\}, \quad n \geq 2.$$

(4) *Space approachability*:

$$r(x_0(k), x_i(k)) \text{ increases as the distance } |x_0(k) - x_i(k)| \text{ decreases at a time point } k.$$

3.2. Grey prediction (GP)

Conventional prediction or forecasting methods usually need a large amount of data to infer a predicted value. The GP model can deal with incomplete information effectively and requires only four data sets or more [13]. As such, the model can be used to predict how a particular combination of product form (design) elements matches a product image, particularly when the information is available only for a limited number of form elements.

The GP model uses a grey differential model (GM) to generate data series from the original data series of a dynamic system. The data series generated by the GM are converted back to the original data series by a reverse procedure to predict the performance of the system. Since the generated data series are more coherent than the original, the accuracy of the modeling is enhanced. The GM has three basic operations: (1) accumulated generation, (2) inverse accumulated generation, and (3) grey modeling. The accumulated generation operation (AGO) is used to build differential equations. The GM is usually represented as GM(M, N) for dealing with M th-order differential equations with N variables. Since any higher-order differential equation can be transferred into a first-order differential equation, we will use the first-order differential equation in this study.

The GM(1,1), a single variable and first-order grey model, is one of the most frequently used grey prediction models. Its procedure involves the following four steps:

Step 1: Denote the original sequence as

$$x^{(0)} = (x^{(0)}(1), x^{(0)}(2), \dots, x^{(0)}(n)), \quad (3)$$

where $x^{(0)}(i)$ is the time series data at time i ($i = 1, 2, \dots, n$).

Step 2: Generate a new sequence $x^{(1)}$ by the AGO based on the original sequence $x^{(0)}$, where

$$x^{(1)} = (x^{(1)}(1), x^{(1)}(2), \dots, x^{(1)}(n)), \quad (4)$$

$$x^{(1)}(1) = x^{(0)}(1), \quad x^{(1)}(k) = \sum_{i=1}^k x^{(0)}(i). \quad (5)$$

Step 3: Define the first-order differential equation as

$$\frac{dx^{(1)}}{dt} + ax^{(1)} = b. \quad (6)$$

Step 4: Use the least-square method to solve (6) by

$$\hat{x}^{(1)}(k+1) = \left(x^{(0)}(1) - \frac{b}{a}\right) e^{-ak} + \frac{b}{a}, \quad (7)$$

$$\hat{x}^{(0)}(k+1) = \hat{x}^{(1)}(k+1) - \hat{x}^{(1)}(k), \quad (8)$$

where

$$\hat{a} = \begin{bmatrix} a \\ b \end{bmatrix} = (B^T B)^{-1} B^T y_1, \quad (9)$$

$$B = \begin{bmatrix} -0.5(x^{(1)}(1) + x^{(1)}(2)) & 1 \\ -0.5(x^{(1)}(2) + x^{(1)}(3)) & 1 \\ \vdots & \vdots \\ -0.5(x^{(1)}(n-1) + x^{(1)}(n)) & 1 \end{bmatrix}, \quad (10)$$

$$y_1 = (x^{(0)}(2), x^{(0)}(3), \dots, x^{(0)}(n))^T. \quad (11)$$

The $\hat{x}^{(1)}(k+1)$ is the prediction value of $x^{(1)}(k+1)$ and $\hat{x}^{(0)}(k+1)$ is the prediction value of $x^{(0)}(k+1)$ at time $k+1$. We can also use the inverse accumulated generation operation (IAGO) to obtain $\hat{x}^{(0)}(k+1)$ as

$$\hat{x}^{(0)}(k+1) = \left(x^{(0)}(1) - \frac{b}{a}\right) (1 - e^a) e^{-ak}. \quad (12)$$

The GM(1,1) grey model can be extended to the GM(1, N) model [20], first order with N variables $(x_1^{(0)}, x_2^{(0)}, x_3^{(0)}, \dots, x_N^{(0)})$. The differential equation can be defined as

$$\frac{dx_1^{(1)}}{dt} + ax_1^{(1)} = b_1 x_2^{(1)} + b_2 x_3^{(1)} + \dots + b_{N-1} x_N^{(1)} = \sum_{i=2}^N b_{i-1} x_i^{(1)}, \quad (13)$$

where $a, b_1, b_2, \dots, b_{N-1}$ are unknown parameters and can be calculated by

$$\hat{a} = (a, b_1, b_2, \dots, b_{N-1}) = (B^T B)^{-1} B^T y_N, \quad (14)$$

where

$$B = \begin{bmatrix} -0.5(x_1^{(1)}(1) + x_1^{(1)}(2)) & x_2^{(1)}(2) & \dots & x_N^{(1)}(2) \\ -0.5(x_1^{(1)}(2) + x_1^{(1)}(3)) & x_2^{(1)}(3) & \dots & x_N^{(1)}(3) \\ \vdots & \vdots & \ddots & \vdots \\ -0.5(x_1^{(1)}(n-1) + x_1^{(1)}(n)) & x_2^{(1)}(n) & \dots & x_N^{(1)}(n) \end{bmatrix}, \quad (15)$$

$$y_N = (x^{(0)}(2), x^{(0)}(3), x^{(0)}(4), \dots, x^{(0)}(n))^T. \quad (16)$$

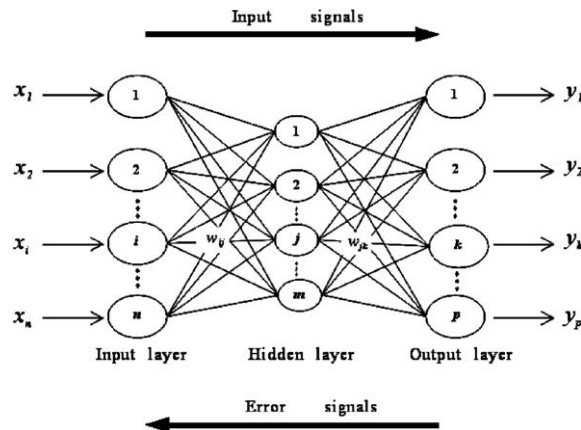


Fig. 2. Three-layer feedforward NN.

The prediction of $x_1^{(1)}$ is defined as

$$\hat{x}_1^{(1)}(k+1) = \left(x_1^{(0)}(1) - \sum_{i=2}^N \frac{b_{i-1}}{a} x_i^{(1)}(k+1) \right) e^{-ak} + \sum_{i=2}^N \frac{b_{i-1}}{a} x_i^{(1)}(k+1). \quad (17)$$

The $\hat{x}^{(1)}(k+1)$ is the prediction value of $x^{(1)}(k+1)$ of the GM(1, N) at time $k+1$.

3.3. Neural networks (NNs)

NNs are non-linear models and are widely used to examine the complex relationship between input variables and output variables [14]. NNs are well suited to formulate the product design process for matching the product form (the input) to the consumer's perception of product image (the output), which is often a black box and cannot be precisely described. Due to their effective learning ability, NNs have been applied successfully in a wide range of fields, using various learning algorithms [6,11,12,24–28]. In this study, we use the multilayered feedforward NNs trained with the backpropagation learning algorithm, as it is an effective and the most popular supervised learning algorithm [14].

As shown in Fig. 2, a typical three-layer network consists of an input layer, an output layer, and one hidden layer, with n , m , and p neurons, respectively (indexed by i , j , and k , respectively) [29]. w_{ij} and w_{jk} represent the weights for the connection between neuron i ($i = 1, 2, \dots, n$) and neuron j ($j = 1, 2, \dots, m$), and between neuron j ($j = 1, 2, \dots, m$) and neuron k ($k = 1, 2, \dots, p$), respectively. In training the network, a set of input patterns or signals, (x_1, x_2, \dots, x_n) , is presented to the network input layer. The network then propagates the inputs from layer to layer until the outputs are generated by the output layer. This involves the generation of the outputs (y_j) of the neurons in the hidden layer as given in (18) and the outputs (y_k) of the neurons in the output layer as given in (19):

$$y_j = f \left(\sum_{i=1}^n x_i w_{ij} - \theta_j \right), \quad (18)$$

$$y_k = f \left(\sum_{j=1}^m x_j w_{jk} - \theta_k \right), \quad (19)$$

where $f(\cdot)$ is the sigmoid activation function as given in (20), and θ_j and θ_k are threshold values.

$$f(X) = \frac{1}{1 + e^{-X}} \quad (20)$$

If the outputs (y_k) generated by (19) are different from the target outputs (y_k^*), errors (e_1, e_2, \dots, e_p) are calculated by (21) and then propagated backwards from the output layer to the input layer in order to update the weights for reducing the errors:

$$e_k = y_k^* - y_k. \quad (21)$$

The weights (w_{jk}) at the output neurons are updated as $w_{jk} + \Delta w_{jk}$, where Δw_{jk} is computed by (known as the delta rule)

$$\Delta w_{jk} = \alpha y_j \delta_k, \quad (22)$$

where α is the learning rate (usually $0 < \alpha \leq 1$) and δ_k is the error gradient at neuron k , given as

$$\delta_k = y_k(1 - y_k)e_k. \quad (23)$$

The weights (w_{ij}) at the hidden neurons are updated as $w_{ij} + \Delta w_{ij}$, where Δw_{ij} is calculated by

$$\Delta w_{ij} = \alpha x_i \delta_j, \quad (24)$$

where α is the learning rate (usually $0 < \alpha \leq 1$) and δ_j is the error gradient at neuron j , given as

$$\delta_j = y_j(1 - y_j) \sum_{k=1}^p \delta_k w_{jk}. \quad (25)$$

The training process is repeated until a specified error criterion is satisfied.

4. Experimental analysis of grey system and NN models

In this section, we present the results of applying the GRA, GP, and NN techniques in order to answer the research questions, using the experimental results summarized in Fig. 1, and Tables 1 and 2.

4.1. GRA model

The GRA model calculates the grey relational degree between each comparison series x_i and the reference series x_0 , using (1) and (2). In this study, the comparison series x_i ($i = 1, 2, \dots, 9$) are the nine form elements, whose values are given in columns 2–10 of Table 2. The reference series x_0 is the average S–C value, as given in the last column of Table 2. The procedure of the GRA involves

the following seven steps:

Step 1: Represent the original data series as

$$\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_9 \end{pmatrix} = \begin{pmatrix} 4.467, & 3.313, & 2.867 & \cdots & 3.733 \\ 2, & 2, & 1, & \cdots & 2 \\ 2, & 3, & 3, & \cdots & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1, & 3, & 1, & \cdots & 2 \end{pmatrix}.$$

Step 2: Normalize the data series.

Step 3: Calculate $|x_0(k) - x_i(k)|$.

Step 4: Calculate $\max\{\max|x_0(k) - x_i(k)|\}$ and $\min\{\min|x_0(k) - x_i(k)|\}$, and obtain the values of 2.42 and 0.00, respectively.

Step 5: Set the distinguishing coefficient $\xi = 0.5$.

Step 6: Calculate the grey relational coefficient of the data series by (1).

Step 7: Calculate the grey relational degree of the data series by (2), and obtain the value of $r(x_0, x_i)$ as follows:

$$r(x_0, x_1) = 0.811, r(x_0, x_2) = 0.755, r(x_0, x_3) = 0.714, r(x_0, x_4) = 0.794, r(x_0, x_5) = 0.795,$$

$$r(x_0, x_6) = 0.751, r(x_0, x_7) = 0.681, r(x_0, x_8) = 0.758, \text{ and } r(x_0, x_9) = 0.659.$$

The result shows that the “top shape” element (x_1) affects the Simple–Complex image the most, followed by the “function buttons style” element (x_5) and the “length and width ratio of body” element (x_4). This implies that the designers should focus their attention more on these most influential elements, when the objective of designing a mobile phone is to achieve a desirable Simple–Complex image. On the contrary, the designers can pay less attention to the less influential elements such as “outline division style” (x_9), the “screen size” element (x_7) and the “bottom shape” element (x_3), as these elements contribute relatively little to the consumers’ perception of the Simple–Complex image on the mobile phone.

4.2. GP models

The GP is used as a technique for determining the best combination of product form elements for matching a desirable product image. In this study, we develop two GP models, called GP and GRA-GP, respectively. The GP model includes all the nine form elements identified from the experimental study, while the GRA-GP model uses only the six most influential elements resulting from the GRA. The 28 samples in the training set, identified in Fig. 1 and given in Table 2, are used as the data set for building these two models.

4.2.1. The GP model

As a GM(1,9), the GP model uses the nine form elements as the comparison series x_i and the average S–C values as the reference series x_0 . As in the GRA model, the data required for these series are given in Table 2.

To build the GP model, we first obtain a new sequence $x^{(1)}$ for each series using (3)–(5) and the AGO as

$$\begin{pmatrix} x_0^{(1)} \\ x_1^{(1)} \\ x_2^{(1)} \\ \vdots \\ x_9^{(1)} \end{pmatrix} = \begin{pmatrix} 4.467, & 7.600, & 10.467, & \dots & 130.067 \\ 2, & 4, & 5, & \dots & 61 \\ 2, & 5, & 8, & \dots & 60 \\ \dots & \dots & \dots & \dots & \dots \\ 1, & 4, & 5, & \dots & 50 \end{pmatrix}.$$

We then apply (13)–(16) to obtain the parameters of \hat{a} as

$$\hat{a} = \begin{pmatrix} a \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \\ b_9 \end{pmatrix} = \begin{pmatrix} 0.024 \\ -0.372 \\ -0.151 \\ -0.002 \\ 0.287 \\ 0.568 \\ 0.930 \\ 0.765 \\ -0.462 \\ -0.390 \end{pmatrix}.$$

The GP model for predicting the S–C value based on the nine form elements is thus built by (17) as

$$\begin{aligned} \hat{x}_0^{(1)}(k+1) = & [4.467 + 1.552x_1^{(1)}(k+1) + 0.631x_2^{(1)}(k+1) + 0.006x_3^{(1)}(k+1) - 1.197x_4^{(1)}(k+1) \\ & - 2.369x_5^{(1)}(k+1) - 3.876x_6^{(1)}(k+1) - 3.191x_7^{(1)}(k+1) + 1.925x_8^{(1)}(k+1) \\ & + 1.628x_9^{(1)}(k+1)]e^{-0.240k} - 1.552x_1^{(1)}(k+1) - 0.631x_2^{(1)}(k+1) \\ & - 0.006x_3^{(1)}(k+1) + 1.197x_4^{(1)}(k+1) + 2.369x_5^{(1)}(k+1) + 3.876x_6^{(1)}(k+1) \\ & + 3.191x_7^{(1)}(k+1) - 1.925x_8^{(1)}(k+1) - 1.628x_9^{(1)}(k+1). \end{aligned} \quad (26)$$

4.2.2. The GRA–GP model

Based on the result of the GRA model, the GRA–GP model considers only the most influential form elements. In this study, we exclude the bottom shape (x_3), the screen size (x_7), and the outline division style (x_9) elements, because their grey relational degree is smaller than 0.75. As such, the GRA–GP model is a GM(1,6), which uses six form elements (x_1, x_2, x_4, x_5, x_6 , and x_8) as the

comparison series x_i and the average S–C values as the reference series x_0 . Table 2 gives the data required for building the model.

The process of building the GRA–GP model is the same as the GP model, as follows: First, $x^{(1)}$ is obtained as

$$\begin{pmatrix} x_0^{(1)} \\ x_1^{(1)} \\ x_2^{(1)} \\ \vdots \\ x_6^{(1)} \end{pmatrix} = \begin{pmatrix} 4.467, & 7.600, & 10.467, & \dots & 130.067 \\ 2, & 4, & 5, & \dots & 61 \\ 2, & 5, & 8, & \dots & 60 \\ \dots & \dots & \dots & \dots & \dots \\ 2, & 4, & 7, & \dots & 58 \end{pmatrix}.$$

Then the parameters of \hat{a} is obtained as

$$\hat{a} = \begin{pmatrix} a \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{pmatrix} = \begin{pmatrix} 0.216 \\ 0.001 \\ -0.200 \\ -0.160 \\ 0.702 \\ 0.917 \\ -0.218 \end{pmatrix}.$$

The GRA–GP model for predicting the S–C value based on the six form elements is thus built as

$$\begin{aligned} \hat{x}_0^{(1)}(k+1) = & [4.467 - 0.001x_1^{(1)}(k+1) + 0.927x_2^{(1)}(k+1) + 0.739x_3^{(1)}(k+1) \\ & - 3.248x_4^{(1)}(k+1) - 4.245x_5^{(1)}(k+1) + 1.008x_6^{(1)}(k+1)]e^{-0.216k} \\ & + 0.001x_1^{(1)}(k+1) - 0.927x_2^{(1)}(k+1) - 0.739x_3^{(1)}(k+1) + 3.248x_4^{(1)}(k+1) \\ & + 4.245x_5^{(1)}(k+1) - 1.008x_6^{(1)}(k+1). \end{aligned} \quad (27)$$

With the GP model in (26) or the GRA–GP model in (27), designers can input the value of the corresponding form elements, and then obtain a predicted S–C value.

4.3. NN models

To examine whether the NN model is an effective technique for determining the best combination of product form elements for matching a desirable product image, we develop two NN models, called NN and GRA–NN, respectively. The NN model uses all the nine form elements identified from the experimental study, while the GRA–NN model uses the result of GRA as a basis for constructing the NN model. As such, like the GRA–GP model, the GRA–NN model excludes the bottom shape (x_3), the screen size (x_7), and the outline division style (x_9) elements.

4.3.1. Building the NN and GRA–NN models

The NN model has 27 input neurons, which are the whole 27 types of the nine form elements given in Table 1. The GRA–NN model has 18 input neurons, which are the 18 types of the six most influential form elements. For both models, if a mobile phone has a particular form element type, the value of the corresponding input neuron is 1; otherwise the value is 0. The output neuron of both models is the S–C value, ranging between 1 and 7, as specified in the experimental study.

In this study, we apply the following four most widely used rules [14] to determine the number of hidden neurons in the single hidden layer for both models:

$$(HN1) \text{ (The number of input neurons + the number of output neurons)}/2, \quad (28)$$

$$(HN2) \text{ (The number of input neurons * the number of output neurons)}^{0.5}, \quad (29)$$

$$(HN3) \text{ (The number of input neurons + the number of output neurons)}, \quad (30)$$

$$(HN4) \text{ (The number of input neurons + the number of output neurons) * 2}. \quad (31)$$

To distinguish between the NN and GRA–NN models using different rules, both models are associated with the rule used, such as -HN1, -HN2, -HN3, or -HN4, as shown in Table 3. Table 3 shows the neurons of the two NN models, including the input layer, the hidden layer, and the output layer.

4.3.2. Training the NN and GRA–NN models

The 28 samples in the training set (given in Table 2 and used for building GP and GPA–GP models) were used to train the NN and GRA–NN models. These models learned fast with the root of mean square (RMS) errors being decreased significantly when these models were trained for about 4000–5000 epochs. For example, the RMS error of the NN–HN1 model decreased from 0.3768 to 0.0984 as the training proceeded for about 4300 epochs. For the training epochs between 4300 and 5000, the error changed slightly within the range between 0.0712 and 0.1010. In a sense, this could

Table 3
Neurons of the NN and GRA–NN models

<i>The NN model</i>	Input layer: 27 neurons for 27 types of the 9 form elements. Output layer: 1 neuron for the S–C value.
-HN1	Hidden layer: 14 neurons, $(27 + 1)/2 = 14$.
-HN2	Hidden layer: 5 neurons, $(27 \times 1)^{0.5} = 5.20 \approx 5$.
-HN3	Hidden layer: 28 neurons, $(27 + 1) = 28$.
-HN4	Hidden layer: 56 neurons, $(27 + 1)2 = 56$.
<i>The GRA–NN model</i>	Input layer: 18 neurons for 18 types of the 6 form elements. Output layer: 1 neuron for the S–C value.
-HN1	Hidden layer: 10 neurons, $(18 + 1)/2 = 9.5 \approx 10$.
-HN2	Hidden layer: 4 neurons, $(18 \times 1)^{0.5} = 4.24 \approx 4$.
-HN3	Hidden layer: 19 neurons, $(18 + 1) = 19$.
-HN4	Hidden layer: 38 neurons, $(18 + 1)2 = 38$.

Table 4
RMS errors of the NN and GRA–NN models for the training set

Number of epochs	The NN model				The GRA–NN model			
	-HN1	-HN2	-HN3	-HN4	-HN1	-HN2	-HN3	-HN4
5000	0.0755	0.1181	0.1180	0.1135	0.0940*	0.1159	0.1122	0.1438
10,000	0.1115	0.1175	0.1423	0.0862	0.1464	0.1220	0.1155	0.1436
15,000	0.1055	0.0924	0.0965	0.0767	0.1139	0.1306	0.1202	0.1294
20,000	0.0617	0.0612	0.0681	0.1068	0.1000	0.0909	0.1460	0.0955
25,000	0.0956	0.0701	0.0622*	0.0985	0.0938	0.1006	0.1271	0.1113
30,000	0.0575	0.0457*	0.0740	0.0834	0.0997	0.0841*	0.1015	0.1321
35,000	0.0581	0.0676	0.1046	0.1026	0.1033	0.1010	0.1156	0.0843
40,000	0.0707	0.0488	0.0975	0.0648*	0.0970	0.0986	0.0925*	0.1173
45,000	0.0430*	0.0662	0.0951	0.0939	0.0941	0.1122	0.1058	0.0768*
50,000	0.0658	0.0611	0.0709	0.1004	0.1034	0.1096	0.1204	0.1000

be regarded as having converged. To find the best training result, we continued the training process up to 50,000 epochs for all models. Table 4 shows some representative results, where the lowest RMS error of each model using a given hidden neuron number is asterisked.

As shown in Table 4, the RMS error of the NN model using the HN1 rule in (28) is the lowest (0.0430), as compared to the other three rules. However, the lowest RMS error (0.0768) of the GRA–NN model is the one using the HN4 rule in (31). This result reflects the notion that there is no best rule for determining the number of neurons of the hidden layer.

In Table 4, the RMS errors of the NN model using different hidden neurons are all smaller than the GRA–NN model. To examine the performance of other simplified NN models, we build the following four simplified NN models by excluding some form elements for the input layer, based on the GRA: (a) GRA–NN-a with 23 input neurons excluding the most influential form element x_1 (GRA value = 0.881); (b) GRA–NN-b with 20 input neurons excluding the two most influential form elements x_1 (0.881) and x_5 (0.795); (c) GRA–NN-c with 24 input neurons excluding the least influential form element x_9 (0.659); and (d) GRA–NN-d with 21 input neurons excluding the two least influential form elements x_9 (0.659) and x_7 (0.681). Like the NN and GRA–NN models, each simplified NN model uses the four rules (HN1, HN2, HN3, and HN4), respectively, for determining the hidden neurons. Table 5 shows some representative results about the corresponding RMS errors of these models after being trained for 5000–50,000 epochs.

Table 5 suggests that the best rule for determining the number of hidden neurons is not consistent among these simplified models. The RMS errors of the simplified models in Table 5 are greater than that of the NN model in Table 4. This result is in line with the general understanding that the more the input neurons, the better the training result.

5. Performance evaluation of grey prediction and NN models

To evaluate the performance of the models developed in this study in terms of their prediction ability in determining the best design combination of mobile phone form elements for matching a

Table 5
RMS errors of simplified NN models for the training set

Number of epochs	The GRA–NN-a model				The GRA–NN-b model			
	-HN1	-HN2	-HN3	-HN4	-HN1	-HN2	-HN3	-HN4
5000	0.1366	0.1376	0.1377	0.1690	0.1227	0.2067	0.1093	0.1842
10,000	0.1280	0.1435	0.0977*	0.1474	0.0993	0.0764*	0.1505	0.1384
15,000	0.1545	0.0978	0.1453	0.0787*	0.1032	0.0776	0.1523	0.1088
20,000	0.1349	0.1229	0.1150	0.0993	0.1198	0.1502	0.0970	0.0951*
25,000	0.1069	0.0871	0.1136	0.1247	0.1393	0.1218	0.1172	0.1062
30,000	0.1230	0.0659	0.1277	0.1030	0.0858	0.1123	0.0747*	0.1001
35,000	0.1247	0.0687	0.1027	0.0842	0.1188	0.1502	0.0917	0.1131
40,000	0.0882*	0.0782	0.1270	0.0812	0.1085	0.1209	0.1023	0.1290
45,000	0.1012	0.0765	0.1232	0.1031	0.1021	0.1099	0.0852	0.1042
50,000	0.0929	0.0615*	0.1066	0.0889	0.0748*	0.1140	0.0933	0.1112

Number of epochs	The GRA–NN-c model				The GRA–NN-d model			
	-HN1	-HN2	-HN3	-HN4	-HN1	-HN2	-HN3	-HN4
5000	0.1127	0.1099	0.0953	0.0774*	0.1159	0.1612	0.1380	0.1146
10,000	0.1116	0.1267	0.1520	0.1465	0.1233	0.0903	0.1401	0.1438
15,000	0.1198	0.0963*	0.1180	0.1147	0.1156	0.1086	0.1095	0.1184
20,000	0.0968*	0.1186	0.0793	0.1441	0.1166	0.1088	0.1055	0.1160
25,000	0.1148	0.1049	0.0898	0.0980	0.1163	0.0917	0.0720*	0.1100
30,000	0.1030	0.1072	0.0984	0.1176	0.1110	0.0620*	0.1143	0.0757*
35,000	0.0982	0.1144	0.0877	0.1172	0.1130	0.0633	0.0893	0.0985
40,000	0.0983	0.1178	0.0929	0.0988	0.1006*	0.0845	0.1102	0.1052
45,000	0.1163	0.0966	0.0643	0.0867	0.1103	0.0666	0.0975	0.0994
50,000	0.1049	0.1060	0.0562*	0.1129	0.1023	0.0887	0.1066	0.1080

given S–C image, the five samples in the test set identified in Fig. 1 are used. The 15 subjects of the third expert group are involved in the process, using the semantic difference method [18] with a 7-point scale (1–7). The second row of Table 6 shows the average S–C values of the five test samples evaluated by the 15 subjects, which are used as a comparison base for the performance evaluation. With the five test samples as the input, Table 6 shows the corresponding S–C values predicted by using the GP, GRA–GP, NN and GRA–NN models, respectively. The last two columns of Table 6 show the (average) RMS errors of these models in comparison with the evaluated S–C values.

Table 6 shows that all the NN-based models outperform the GP-based models in terms of the RMS error, suggesting that the NN model is a better technique for answering the research questions identified in this study. In addition, it is noteworthy that the RMS errors of the GRA–NN-a and GRA–NN-b models are greater than the GRA–NN-c and GRA–NN-d models. This indicates that the performance of the NN-based model decreases much more, if the more influential form elements are excluded other than the less influential ones.

To examine whether the performance of the NN-based models in Table 6 differ significantly, we perform the test of statistical significance on all the models. Table 7 shows the result of the multiple

Table 6

Predicted S–C values and RMS errors of all models for the test set

	Phone no.					RMS errors	
	14	23	29	33	45		
Evaluated S–C value	6.400	6.600	2.733	4.533	4.600		
The GP model	4.021	8.819	7.313	4.857	4.170		0.361
The GRA–GP model	4.388	9.910	3.343	2.229	6.392		0.312
<i>The NN model</i>							
-HN1	6.529	7.158	4.227	5.364	2.184	0.1839	0.178
-HN2	5.960	6.969	4.107	5.275	1.967	0.1899	
-HN3	6.437	7.299	4.310	5.262	2.655	0.1647	
-HN4	6.562	7.313	4.392	5.303	2.592	0.1715	
<i>The GRA–NN model</i>							
-HN1	4.865	5.596	4.643	4.420	2.203	0.2179	0.213
-HN2	5.203	5.447	4.810	4.503	2.377	0.2114	
-HN3	4.877	5.494	4.582	4.408	2.185	0.2183	
-HN4	5.310	5.426	4.506	4.109	2.296	0.2041	
<i>The GRA–NN-a model</i>							
-HN1	4.116	5.860	6.651	5.098	5.394	0.2864	0.283
-HN2	4.920	7.520	6.348	5.910	5.636	0.2664	
-HN3	3.578	6.475	6.582	5.037	4.910	0.2934	
-HN4	3.904	6.466	6.600	5.082	5.132	0.2847	
<i>The GRA–NN-b model</i>							
-HN1	4.674	6.279	6.650	3.608	6.587	0.2940	0.286
-HN2	4.067	6.068	6.553	4.381	5.459	0.2800	
-HN3	4.196	5.848	6.374	4.157	6.446	0.2875	
-HN4	3.813	5.895	6.393	4.182	5.326	0.2810	
<i>The GRA–NN-c model</i>							
-HN1	4.973	5.927	4.758	4.586	2.420	0.2054	0.217
-HN2	4.835	5.867	4.617	4.433	2.343	0.2081	
-HN3	5.816	5.564	5.803	3.466	2.501	0.2469	
-HN4	4.835	6.087	4.911	4.613	2.551	0.2083	
<i>The GRA–NN-d model</i>							
-HN1	4.851	5.862	4.685	4.447	2.336	0.2102	0.217
-HN2	4.322	5.485	4.380	5.118	2.412	0.2233	
-HN3	4.826	5.654	4.699	4.555	2.260	0.2174	
-HN4	4.752	5.407	4.517	4.508	2.336	0.2152	

comparisons by using the least significant difference (LSD) test. In Table 7, the asterisk indicates that the performance of the corresponding pair of models differs significantly at the 0.05 level. Except for GRA–NN, GRA–NN-c, and GRA–NN-d, all models have significantly different performance. The NN model performs best.

Table 7

Statistical significance test of the performance of NN based models on the test set

Multiple comparisons	Mean difference	<i>p</i> value	95% confidence interval	
			Lower bound	Upper bound
NN model vs. GRA–NN model	–0.0354*	0.000	–0.0523	–0.0185
GRA–NN-a model	–0.1052*	0.000	–0.1221	–0.0883
GRA–NN-b model	–0.1081*	0.000	–0.1250	–0.0912
GRA–NN-c model	–0.0397*	0.000	–0.0566	–0.0228
GRA–NN-d model	–0.0390*	0.000	–0.0559	–0.0221
GRA–NN model vs. GRA–NN-a model	–0.0698*	0.000	–0.0867	–0.0529
GRA–NN-b model	–0.0727*	0.000	–0.0896	–0.0558
GRA–NN-c model	–0.0043	0.603	–0.0211	0.0126
GRA–NN-d model	–0.0036	0.660	–0.0205	0.0133
GRA–NN-a model vs. GRA–NN-b model	–0.0029	0.772	–0.0198	0.0140
GRA–NN-c model	0.0656*	0.000	0.0487	0.0824
GRA–NN-d model	0.0662*	0.000	0.0493	0.0831
GRA–NN-b model vs. GRA–NN-c model	0.0685*	0.000	0.0516	0.0853
GRA–NN-d model	0.0691*	0.000	0.0522	0.0860
GRA–NN-c model vs. GRA–NN-d model	0.0007	0.936	–0.0162	0.0175

To further examine the prediction performance of the NN model, we conduct an experimental test on the 21 unselected mobile phones in the out-of-sample test set. Table 8 shows the Kansei evaluation matrix for the out-of-sample test set. Table 9 shows the model prediction performance of NN based models on the out-of-sample test set. Table 10 shows the significance test result of the multiple comparisons among all the results in Table 9.

The results in Tables 9 and 10 show that the NN, GRA–NN, GRA–NN-c, and GRA–NN-d models perform well and have no significant performance difference among them. These models perform significantly better than the GRA–NN-a and GRA–NN-b models. This implies that excluding less influential form elements from a NN-based model, according to the GRA model, may have no impact on the prediction performance of the model. In addition, it is noteworthy that the RMS error of the NN model for the out-of-sample test set in Table 9 is higher than that for the test set in Table 6. This is because there are only five mobile phones in the test set, while the out-of-sample test set has 21 mobile phones. If we select any five mobile phones from the out-of-sample test set for testing, the RMS error ranges from 0.1657 to 0.1922, with an average of 0.183. As shown in Table 6, the RMS error of the four NN models for the five-sample test set falls into this range. This indicates that the NN model performs equally well on the test set and on the out-of-sample test set.

The above evaluation results suggest that the NN-based model should be used to help product designers determine the best combination of mobile phone form elements for achieving a desirable product image. In situations where product designers would like to focus on fewer form elements in the design process, the result of GRA model can be used to simplify the NN model, without greatly compromising its performance.

Table 8

Kansei evaluation matrix for the 21 mobile phones in the out-of-sample test set

Phone no.	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	S–C value			
										Average	Min	Max	Standard deviation
1	1	3	1	3	1	1	1	1	1	5.800	2	7	1.681
3	1	1	1	2	2	1	3	1	2	3.600	1	7	2.091
5	1	3	1	2	2	1	1	1	1	3.333	1	7	2.150
6	2	1	2	3	1	1	1	1	1	6.667	4	7	0.789
8	2	3	2	3	2	2	2	3	1	3.800	2	7	1.720
9	1	2	1	2	2	2	2	2	1	5.800	4	7	1.275
11	2	3	1	2	2	1	3	3	1	4.000	1	7	1.932
13	3	3	2	2	1	1	3	1	1	3.200	1	6	1.796
15	2	1	1	3	1	1	1	2	1	6.133	4	7	1.204
17	3	2	3	1	1	1	2	3	1	4.000	1	6	1.862
20	1	3	2	2	1	2	1	2	3	6.067	3	7	1.289
21	2	2	1	2	3	1	2	3	1	5.600	4	7	1.254
28	3	2	3	3	1	1	2	2	1	5.867	4	7	1.147
32	2	3	2	2	1	1	2	3	1	5.333	1	7	1.738
37	2	2	2	2	1	1	2	1	1	3.400	1	5	1.451
40	3	3	2	2	3	1	2	1	3	4.267	2	6	1.482
43	2	2	2	1	3	2	1	2	3	4.267	2	6	1.526
47	2	3	1	3	1	1	1	3	1	5.867	4	7	0.884
49	3	3	2	1	1	2	2	3	2	5.267	2	7	1.731
53	3	1	3	2	2	1	1	2	1	3.600	1	7	1.890
54	2	3	1	3	1	2	2	3	1	5.333	3	7	1.738

Table 9

Model prediction performance on the out-of-sample test set

Models	RMS errors				
	-HN1	-HN2	-HN3	-HN4	Average
The NN model	0.2266	0.2172	0.2367	0.2328	0.227
The GRA–NN model	0.2458	0.2260	0.2324	0.2384	0.236
The GRA–NN-a model	0.3099	0.3073	0.3176	0.3196	0.314
The GRA–NN-b model	0.3117	0.3161	0.3100	0.3157	0.313
The GRA–NN-c model	0.2313	0.2345	0.2169	0.2242	0.228
The GRA–NN-d model	0.2249	0.2422	0.2347	0.2348	0.234

The NN model built in this study can be used, in conjunction with the GRA model if needed, to work out the best combination of the form elements for a desirable product image specified by product designers. Each possible combination of nine form elements in Table 1 can be inputted to the NN model to obtain an S–C value. As a result, a form design database consisting of 17,496 ($=4 \times 3 \times 3 \times 3 \times 3 \times 2 \times 3 \times 3 \times 3$) different combinations of mobile phone form elements together with










Table 10

Statistical significance test of model performance on the out-of-sample test set

Multiple comparisons	Mean difference	<i>p</i> value	95% confidence interval	
			Lower bound	Upper bound
NN model vs. GRA–NN model	−0.0083	0.160	−0.0178	0.0032
GRA–NN-a model	−0.0863*	0.000	−0.0958	−0.0748
GRA–NN-b model	−0.0861*	0.000	−0.0955	−0.0746
GRA–NN-c model	−0.0016	0.753	−0.0089	0.0121
GRA–NN-d model	−0.0068	0.259	−0.0163	0.0047
GRA–NN model vs. GRA–NN-a model	−0.0780*	0.000	−0.0884	−0.0675
GRA–NN-b model	−0.0777*	0.000	−0.0882	−0.0672
GRA–NN-c model	0.0079	0.091	−0.0016	0.0194
GRA–NN-d model	0.0015	0.767	−0.0090	0.0120
GRA–NN-a model vs. GRA–NN-b model	0.0002	0.965	−0.0103	0.0107
GRA–NN-c model	0.0859*	0.000	0.0764	0.0974
GRA–NN-d model	0.0795*	0.000	0.0690	0.0900
GRA–NN-b model vs. GRA–NN-c model	0.0857*	0.000	0.0762	0.0972
GRA–NN-d model	0.0792*	0.000	0.0684	0.0897
GRA–NN-c model vs. GRA–NN-d model	−0.0064	0.155	−0.0179	0.0031

Table 11

The best combination of product form elements for the S–C value of 2

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
								

their associated S–C values can be built. The form design database can then be used to determine which form combination(s) will best match a given S–C value. To illustrate, Table 11 shows the best form combination for the S–C value of 2 (e.g. image of “quite simple”), as its S–C value is 2.083, which is the closest among all form combinations. This result can be incorporated into a computer aid design (CAD) system [4] to build a 3D model for facilitating the design process of mobile phones, as shown in Fig. 3.

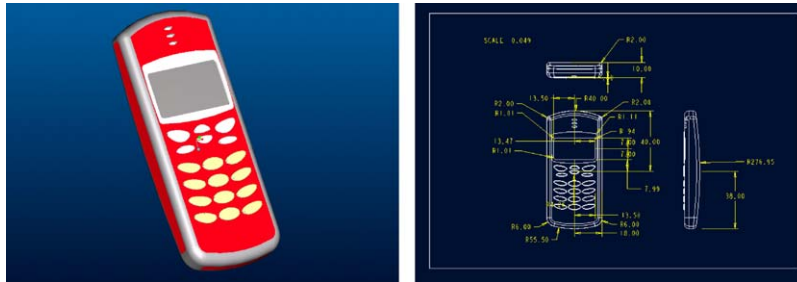


Fig. 3. 3D model of the CAD system.

6. Conclusion

In this paper, we have presented a new approach for examining the ideal design of product form for a given product image, with an experimental study on mobile phones. We have used the concept of Kansei Engineering to extract 33 experimental samples of mobile phones, with 27 form types categorized into nine form elements. We have performed the grey relational analysis (GRA) to find out the most influential form elements of mobile phones for the simple–complex image. To determine how the product form elements can be best combined to match a desirable product image, we have developed grey prediction (GP)-based and NN-based models. The experimental result of evaluating these models has suggested that NN-based models have a better prediction performance and should be used to determine the best form design for matching a given S–C image. The GRA model can be used to simplify the NN model for helping product designers focus on more influential form elements. The result of this study provides useful insights for designing form elements of a product for enhancing a particular image of the product. Although the mobile phones are chosen as an illustration of the approach, the approach can be applied to other products with various design elements.

Acknowledgements

This research was supported in part by the National Science Council of Taiwan, ROC under Grant No. NSC90-2218-E-006-032. We are grateful to the 35 product design experts in Taiwan for their participation and assistance in the experimental study. We also thank Professor Kate Smith, the guest editor, and anonymous referees for their valuable comments and advice.

References

- [1] Chuang MC, Chang CC, Hsu SH. Perceptual elements underlying user preferences toward product form of mobile phones. *International Journal of Industrial Ergonomics* 2001;27:247–58.
- [2] Chuang MC, Ma YC. Expressing the expected product images in product design of micro-electronic products. *International Journal of Industrial Ergonomics* 2001;27(4):233–45.
- [3] Hsiao SW, Chen CH. A semantic and shape grammar based approach for product design. *Design Studies* 1997;18(3):275–96.

- [4] Hsiao SW, Liu MC. A morphing method for shape generation and image prediction in product design. *Design Studies* 2002;23(6):533–56.
- [5] Jindo T, Hirasago K, Nagamachi M. Development of a design support system for office chairs using 3-D graphics. *International Journal of Industrial Ergonomics* 1995;15(1):49–62.
- [6] Hsu CH, Jiang BC, Lee ES. Fuzzy neural network modeling for product development. *Journal of Mathematical and Computer Modeling* 1999;29:71–81.
- [7] Nagamachi M. Kansei Engineering as a powerful consumer-oriented technology for product development. *Applied Ergonomics* 2002;33:289–94.
- [8] Nagamachi M. Kansei Engineering: a new ergonomics consumer-oriented technology for product development. *International Journal of Industrial Ergonomics* 1995;15(1):3–11.
- [9] Fukushima K, Kawata H. Human sensory perception oriented image processing in a color copy system. *International Journal of Industrial Ergonomics* 1995;15(1):63–74.
- [10] Kashiwagi K, Matsubara Y, Nagamachi M. A feature detection mechanism of design in Kansei Engineering. *Human Interface* 1994;9(1):9–16.
- [11] Ishihara S, Ishihara K, Nagamachi M. An automatic builder for a Kansei Engineering expert system using self-organizing neural networks. *International Journal of Industrial Ergonomics* 1995;15(1):25–37.
- [12] Guan SS, Lin YC. A study on the color and style collocation of mobile phones using neural network method. *Journal of the Chinese Institute of Industrial Engineers* 2001;18(6):84–94.
- [13] Deng JL. Control problems of grey system. *System and Control Letters* 1-1982; 288–94.
- [14] Nelson M, Illingworth WT. A practical guide to neural nets. New York: Addison-Wesley; 1991.
- [15] Lin YC, Lai HH, Guan SS. A study of form and color collocation of mobile phones using Kansei methods. *Proceedings of International Conference on Advanced Industrial Design (ICAID)*, Taiwan, 2001. p. 137–42.
- [16] Cross M. *Engineering design methods*. London: Wiley; 1994.
- [17] Nielsen J. *Usability engineering*. New York: AP Professional; 1993.
- [18] Osgood CE, Suci CJ. *The measurement of meaning*. Urbana: University of Illinois Press; 1957.
- [19] Hsu CC, Chen CY. Applications of improved grey prediction model for power demand forecasting. *Energy Conversion and Management* 2003;44:2241–9.
- [20] Tseng FM, Yu HC, Tzeng GH. Applied hybrid grey model to forecast seasonal time series. *Technological Forecasting and Social Change* 2001;67:291–302.
- [21] Chang TC, Lin SJ. Grey relation analysis of carbon dioxide emissions from industrial production and energy uses in Taiwan. *Journal of Environment Management* 1999;56:247–57.
- [22] Hu WB, Hua B, Yang CZ. Building thermal process analysis with grey system method. *Building and Environment* 2002;37:599–605.
- [23] Hu WB, Yang CZ. Grey model of direct solar radiation intensity on the horizontal plane for cooling loads calculation. *Building and Environment* 2000;35:587–93.
- [24] Smith K, Palaniswami M, Krishnamoorthy M. A hybrid neural approach to combinatorial optimization. *Computers and Operations Research* 1996;23(6):597–610.
- [25] Kwok T, Smith K. Experimental analysis of chaotic neural network models for combinatorial optimization under a unifying framework. *Neural Networks* 2000;13(7):731–44.
- [26] Smith K, Gupta J. Neural networks in business: techniques and applications for the operations researcher. *Computers and Operations Research* 2000;27:1023–44.
- [27] Wei CH. Analysis of artificial neural network models for freeway ramp metering control. *Artificial Intelligence in Engineering* 2001;15:241–52.
- [28] Liu PX, Zuo MJ, Meng M. Using neural network function approximation for ideal design of continuous-state parallel-series systems. *Computers and Operations Research* 2003;30:339–52.
- [29] Negnevitsky M. *Artificial intelligence*. New York: Addison-Wesley; 2002.