Multiclass SVM-RFE for product form feature selection

Meng-Dar Shieh *, Chih-Chieh Yang

Department of Industrial Design, National Cheng Kung University, No. 1, University Road, Tainan, Taiwan 70101, ROC

Abstract

Various form features affect consumer preference regarding product design. It is, therefore, important that designers identify these critical form features to aid them in developing appealing products. However, the problems inherent in choosing product form features have not yet been intensively investigated. In this paper, an approach based on multiclass support vector machine recursive feature elimination (SVM-RFE) is proposed to streamline the selection of optimum product form features. First, a one-versus-one (OVO) multiclass fuzzy support vector machines (multiclass fuzzy SVM) model using a Gaussian kernel was constructed based on product samples from mobile phones. Second, an optimal training model parameter set was determined using two-step cross-validation. Finally, a multiclass SVM-RFE process was applied to select critical form features by either using overall ranking or class-specific ranking. The weight distribution of each iterative step can be used to analyze the relative importance of each of the form features. The results of our experiment show that the multiclass SVM-RFE process is not only very useful for identifying critical form features with minimum generalization errors but also can be used to select the smallest feature subset for building a prediction model with a given discrimination capability.

© 2007 Elsevier Ltd. All rights reserved.

Keywords: Feature selection; Multiclass support vector machines recursive feature elimination (SVM-RFE); Mobile phone design

1. Introduction

The way that a product looks is one of the most important factors affecting a consumer’s purchasing decision. Traditionally, the success of a product’s design depended on the designers’ artistic sensibilities, which quite often did not meet with great acceptance in the marketplace. Many systematic product form design studies have been carried out to get a better insight into consumer preferences and to develop appealing products in a more effective manner. The most notable research is Kansei Engineering (Jindo, Hirasago, & Nagamachi, 1995). However, the problem of product form feature selection according to consumer preference has not been intensively investigated. Consumer preference is often influenced by a wide variety of form features. The number of form features could be many and might be highly correlated to each other. The relative importance of each of the form features is hard to identify so the selection of the critical form features that please the consumer is a difficult task.

In the product design field, critical design features are often arrived at based on the opinions of expert (such as product designers) or focus groups. However, the selection of features based on expert opinion has its drawbacks such as a lack of objectivity and expert availability (Han & Kim, 2003). Only a few attempts have been made to overcome these shortcomings in the product form feature selection process. Han and Kim (2003) used several traditional statistical methods for screening critical design features including principal component regression (PCR), cluster analysis, and partial least squares (PLS). Han and Yang (2004) applied a genetic algorithm-based partial least squares method (GA-based PLS) to screen design variables.

Actually, the problem of feature selection exists in many other fields besides that of product design. The crux of the problem is how to find the subset of features with the least
possible generalization errors and to select the smallest possible subset with a given discrimination capability. Different approaches have been proposed for solving the feature selection problem including rough sets (Wakaki, Itakura, & Tamura, 2004), rough-fuzzy (Jensen, 2005), neuro-fuzzy (Basak, De, & Pal, 1998; Pal, Basak, & De, 1996), and support vector machines (SVM) (Chen & Lin, 2006; Liu & Zheng, 2006). Of these approaches, SVM’s remarkable and robust performance with respect to sparse and noisy data makes it a first choice for a number of applications. SVM has also provided better performance than traditional learning techniques (Burges, 1998).

Another crucial issue in solving the feature selection problem is how to deal with the correlations between attributes and process their nonlinear properties (Park & Han, 2004; Shimizu & Jindo, 1995). The most widely adapted techniques such as multiple regression analysis (Han, Kim, & Yun, 2004) and multivariate analysis (Jindo et al., 1995) do not handle nonlinear relationships very well. In contrast, SVM is known for its elegance in solving nonlinear problems by applying the “kernels” technique, which automatically maps a feature space nonlinearly. Of the commonly used kernel functions, the Gaussian kernel is favored for many applications due to its good features (Wang, Xu, & Lu, 2003); and thus was adapted for use in this study.

In many real-world applications, input samples may not be exactly assigned to one class and the effects of the training samples might be different. It is more important for some to be fully assigned to one class so that the SVM can better separate these samples. Some samples might be noisy and less meaningful and should be discarded. Treating every data sample equally may cause unsuitable overfitting problems. The original SVM format lacked this kind of ability. Lin and Wang (2002) proposed the fuzzy SVM concept, which combines fuzzy logic and SVM and allows different training samples to make different contributions to their own class. The nub of their concept is to fuzzify the training set and assign each data sample a membership value according to its attitude toward one class. However, their research is limited to binary SVM and does not encompass multiclass SVM. SVM was originally designed for binary classification. In order to extend binary SVM to multiclass SVM, several methods based on binary SVM have been proposed such as “one-versus-rest” (OVR), “one-versus-one” (OVO), and directed acyclic graph SVM (DAGSVM). The OVO and DAGSVM methods were shown to have greater accuracy in practical use (Hsu & Lin, 2001), but the OVO method is adapted in this study.

Based on whether or not feature selection is performed independently of the learning algorithm that constructs the classifier, feature selection approaches can be grouped into two categories: the filter approach and the wrapper approach (Kohavi & John, 1997). The wrapper approach is classifier-dependent. Based on the classification accuracy, the method evaluates directly the “goodness” of the selected feature’s subset, which should intuitively yield a better performance. Many reported experimental results also favored the wrapper approach (Juang & Katagiri, 1992; Kohavi & John, 1997; Wakaki et al., 2004). There were just a few algorithms in the literature that were proposed for feature selection in the context of SVM (Bradley & Mangasarian, 1998; Evgeniou, Pontil, Papageorgiou, & Poggio, 2003; Guyon, Weston, Barnhill, & Bapnik, 2002; Mao, 2004). Support vector machine recursive feature elimination (SVM-RFE) was first proposed by Guyon et al. (2002) to aid in gene selection for cancer classification. SVM-RFE is a wrapper approach used in two-class circumstances. It was demonstrated that the features selected by SVM-RFE yielded better classification performance than the other methods mentioned in the study of Guyon et al. (2002).

This study uses an approach based on multiclass SVM-RFE for product form feature selection. The collected form features of product samples were used as input vectors to construct an OVO multiclass fuzzy SVM model using a Gaussian kernel. An optimal training parameter set of the model was determined by two-step cross-validation. The multiclass SVM-RFE process was conducted to select critical form features by either using overall ranking or class-specific ranking. A case study of mobile phone design is given to demonstrate the analysis results. The remainder of the paper is organized as follows: Section 2 reviews the theoretical backgrounds of fuzzy SVM and the multiclass SVM-RFE process for feature selection; Section 3 presents the proposed model for product form feature selection; Section 4 describes the experimental design; Section 5 presents the experimental results and analyses of the proposed model; and Section 6 offers some brief conclusions.

2. Theoretical backgrounds

2.1. Fuzzy support vector machines for binary classification

A set $S$ of $l$ training samples, each represented, are given as $(x_i, y_i, \mu_i)$ where $x_i$ is the feature vector, $y_i$ is the class label, and $\mu_i$ is the fuzzy membership function. These samples are labeled: $y_i \in \{-1, +1\}$, a fuzzy membership value $\sigma < \mu_i \leq 1$ with $i = 1, \ldots, l$, and sufficient small $\sigma > 0$. Data samples where $\mu_i = 0$ mean nothing and can be removed from the training set without affecting the result. Let $z_i = \phi(x_i)$ denote the corresponding feature space vector with a mapping function $\phi$ from $R^N$ to a feature space $Z$. One hyperplane can be defined as:

$$w \cdot z + b = 0$$

The set $S$ is said to be linearly separable if there exists $(w, b)$ so that the inequalities

$$w \cdot z_i + b \geq +1 \Rightarrow y_i = +1$$

$$w \cdot z_i + b \leq -1 \Rightarrow y_i = -1$$

\[(2)\]
are valid for all data samples of the set S. To deal with data that are not linearly separable, the previous analysis can be generalized by introducing some non-negative variables \( \xi_i \geq 0 \) so that Eq. (2) is modified to

\[
y_i(w \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, l
\]

(3)

the non-zero \( \xi_i \) in Eq. (3) are those for which the data samples \( x_i \) does not satisfy Eq. (3). Thus the term \( \sum_{i=1}^{l} \xi_i \) can be thought of as some measure of the number of misclassifications. Since the fuzzy membership value \( \mu_i \) is the attitude of the corresponding sample \( x_i \) toward one class and the parameter \( \xi_i \) is the measure of error in the SVM, the term \( \mu_i \xi_i \) is a measure of error with different weighting. The optimal hyperplane problem is then regarded as the solution to minimize

\[
\frac{1}{2} \|w\|^2 + C \sum_{i=1}^{l} \mu_i \xi_i
\]

subject to

\[
y_i(w \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, l, \quad \xi_i \geq 0, \quad i = 1, \ldots, l
\]

(4)

where \( C \) is a constant. The parameter \( C \) can be regarded as a regulation parameter. Tuning this parameter can make a balance between the minimization of the error function and the maximization of the margin of the optimal hyperplane. It is noted that a smaller \( \mu_i \) reduces the effect of the parameter \( \xi_i \) so that the corresponding point \( x_i \) is treated as less important. The optimization problem (4) can be solved by introducing a Lagrange multiplier \( z \) and transformed into:

minimize

\[
W(x) = \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} a_i a_j y_i y_j (\xi_i \cdot \xi_j) - \sum_{i=1}^{l} a_i
\]

subject to

\[
\sum_{i=1}^{l} y_i a_i = 0, \quad 0 \leq a_i \leq \mu_i C, \quad i = 1, \ldots, l
\]

(5)

and the Kuhn–Tucker conditions are defined as

\[
a_i (y_i (w \cdot z_i + b) - 1 + \xi_i) = 0, \quad i = 1, \ldots, l
\]

\[
(\mu_i C - a_i) \xi_i = 0, \quad i = 1, \ldots, l
\]

(6)

(7)

The data sample \( x_i \) with the corresponding \( a_i > 0 \) is called a support vector. There are two types of support vectors. The one with corresponding \( 0 < a_i < \mu_i C \) lies on the margin of the hyperplane. The one with corresponding \( a_i = \mu_i C \) is misclassified. An important difference between SVM and fuzzy SVM is that the point with the same value of \( a_i \) may indicate a different type of support vector in fuzzy SVM due to the factor \( \mu_i \). (Lin & Wang, 2002). The mapping \( \phi \) is usually nonlinear and unknown. Instead of calculating \( \phi \), the kernel function \( K \) is used to compute the inner product of the two vectors in the feature space \( Z \) and thus implicitly defines the mapping function, which is

\[
K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) = \xi_i \cdot \xi_j
\]

(8)

The following are three types of commonly used kernel functions:

linear kernel: \( K(x_i, x_j) = x_i \cdot x_j \)  (9)

polynomial kernel: \( K(x_i, x_j) = (1 + x_i \cdot x_j)^p \)  (10)

Gaussian kernel: \( K(x_i, x_j) = \exp(-\|x_i - x_j\|^2 / 2\sigma^2) \)  (11)

where the order \( p \) of polynomial kernel in Eq. (10) and the spread with \( \sigma \) of the Gaussian kernel in Eq. (11) are adjustable kernel parameters. The weight vector \( w \) and the decision function can be expressed by using the Lagrange multiplier \( a_i \):

\[
w = \sum_{i=1}^{l} a_i y_i \xi_i
\]

(12)

\[
D(x) = \text{sign}(w \cdot z + b) = \text{sign} \left( \sum_{i=1}^{l} a_i y_i K(x_i, x) + b \right)
\]

(13)

2.2. Feature selection based on multiclass SVM-RFE

In order to adapt fuzzy SVM used for binary classification into one used for a multiclass problem, in this study the OVO method is used (Hsu & Lin, 2001). The OVO method constructs \( \frac{n(n-1)}{2} \) binary SVMs for an \( n \)-class problem, where each of the \( n(n-1)/2 \) SVMs is trained on data samples from two classes. Data samples are partitioned by a series of optimal hyperplanes. This means that the optimal hyperplane training data is maximally distant from the hyperplane itself, and the lowest classification error rate will be achieved when using this hyperplane to classify the current training set. These hyperplanes can be modified from Eq. (1) as

\[
w_{st} \cdot z + b_{st} = 0
\]

(14)

and the decision functions are defined as \( D_{st}(x_i) = w_{st} \cdot z + b_{st} \), \( s \) and \( t \) mean two arbitrary classes separated by an optimal hyperplane in \( n \) classes; \( w_{st} \) is the weight vector and \( b_{st} \) is the bias term. After all \( n(n-1)/2 \) classifiers are constructed, a max-win voting strategy is used to examine all data samples (Krebel, 1999). Each of the \( n(n-1)/2 \) OVO SVMs casts one vote. If \( D_{st}(x_i) \) says \( x_i \) is in the \( s \)th class, then the vote of \( x_i \) for the \( st \)th is added by one. Otherwise, the \( st \)th is increased by one. Then \( x_i \) can be predicted in the class with the largest vote. Since fuzzy SVM is a natural extension of traditional SVM, the same OVO scheme can be used to deal with multiclass problems without any difficulty.

An efficient wrapper approach called SVM-RFE was used to conduct product form feature selection in this study. SVM-RFE is a sequential backward feature elimination method based in binary SVM, which was proposed to select a relevant set of features for cancer classification problem (Guyon et al., 2002). The selection criterion of SVM-RFE was developed according to optimal brain damage (OBD) which has proved to be better than earlier methods (Rakotomamonjy, 2003). OBD was first proposed by LeCun, Denker, Solla, Howard, and Jackel, 1990 and uses the change of the cost function as the feature selection
criterion, which is defined as the second-order term in the Taylor series of the cost function:

\[
c_f = \frac{1}{2} \left[ \tilde{\alpha}_f \right]^2 (D \alpha_f)^2
\]

in which \( L \) is the cost function of any learning machine, and \( \alpha \) is the weight of features. OBD uses \( c_f \) to approximate the change in the cost function caused by removing a given feature \( f \) by expanding the cost function in the Taylor series. Therefore, for binary SVMs, the measure of OBD can be considered as the removed feature that has the least influence on the weight vector norm \( |w|^2 \) in Eq. (5). The ranking criterion can be written as

\[
c_f = \left| \|w\|^2 - \|w^{(f)}\|^2 \right|
\]

\[
= \frac{1}{2} \left[ \sum_{i=1}^{n} \sum_{j=1}^{n} x^*_i x_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{n} \sum_{j=1}^{n} x^{(f)}_i x^{(f)}_j y_i y_j K^{(f)}(x_i, x_j) \right]
\]

where \( x^*_i \) is the corresponding solution of Eq. (5), the notation \(-f\) means that the feature \( f \) has been removed, and \( K \) is the kernel function calculated using \( x_i \) and \( x_j \) (Rakotomamonjy, 2003). To compute the change in an objective function by removing feature \( f, x^{(f)}_i \), is supposed to be equal to in order to reduce computational complexity then re-compute the kernel function \( K^{(f)}(x_i, x_j) \). SVM-RFE starts with all the features. At each step, feature weights are obtained by comparing the training samples with the existing features. Then, the feature with the minimum \( c_f \) is removed. This procedure continues until all features are ranked according to the removed order.

In this study, the binary SVM-RFE was extended to multiclass SVM-RFE. Each feature ranking criterion \( c_{sf} \) of two arbitrary classes \( s \) and \( t \), calculated from \( n(n-1)/2 \) OVO SVMs, was used to calculate the feature ranking in the multiclass fuzzy SVM model. It has been reported that multiclass feature selection has not been widely used in the bioinformatics field due to the expense of calculating data for a large amount of genes (Mao, Zhou, Pi, Sun, & Wong, 2005). Compared to the problems of gene selection, the number of form features and product samples used in this study is relative small so the data for the multiclass feature selection problem can still be computed efficiently. In addition, the binary SVM-RFE for gene selection applications often uses the linear kernel function to accelerate the training process (Duan, Rajapakse, Wang, & Azuaje, 2005). However, the nonlinear kernel function was preferred in this study to deal with the nonlinear relationship between product form features. Moreover, speeding up the SVM-RFE can be accomplished by simultaneously eliminating a number of features instead of only one feature at a time. However, the price for accelerating the process is some degradation in the ranking accuracy (Guyon et al., 2002) and was not done in this study.

### 3. Product form feature selection model

The proposed approach aims to construct a product form feature selection model based on consumer preferences. First, an OVO multiclass fuzzy SVM model using a Gaussian kernel was constructed. Next, a two-step cross-validation was used to search for the best combination of parameters to obtain an optimal training model. Finally, a multiclass SVM-RFE process was conducted (based on the optimal multiclass model) to select the critical form features. The relative importance of the form features can be analyzed during each iterative step.

#### 3.1. Constructing multiclass fuzzy SVM model

In order to construct the multiclass fuzzy SVM model, product samples were collected and their form features systematically examined. Each product sample was assigned a class label and a fuzzy membership function agreeing with this label to formulate a multiclass classification problem. This multiclass problem was then divided into a series of OVO SVM sub-problems. Each OVO SVM used a Gaussian kernel function in Eq. (11) to deal with the nonlinear correlations between product form features. The objective of multiclass classification was to correctly discriminate each of these classes from the others. Each OVO problem was addressed by two different class labels (e.g., sports versus simplicity). Each classifier used the fuzzy SVM to define a hyperplane that best separated the product samples into two classes. Each test sample was sequentially presented to each of the \( n \times (n-1)/2 \) OVO classifiers where it could predict to which label it belonged, based on which of the OVO classifiers had the largest vote.

#### 3.2. Choosing optimal parameters using cross-validation

In order to obtain the best generalization performance and reduce the overfitting problem, five-fold cross-validation was used for choosing the optimal parameters. The whole training samples were randomly divided into five subsets of approximately equal size. Each multiclass model was trained using \( 5 - 1 = 4 \) subsets and tested using the remaining subset. Training was repeated five times and the average testing error rates for all the five subsets that were not included in the training data were calculated. Each binary classifier required the selection of two parameters for the Gaussian kernel, which were the regularization parameter \( C \) and the kernel parameter \( \sigma^2 \) in Eq. (11). The parameters \( C \) and \( \sigma^2 \) of each classifier within the multiclass model were set to be the same for calculation efficiency. Since the process of cross-validation is very time-consuming, a two-step grid search was conducted to find the opti-
nal parameter pair (Hsu, Chang, & Lin, 2003). In the first step, a coarse grid search was made using the following sets of values: $C = \{10^{-3}, \ldots, 10^3\}$ and $\sigma^2 = \{10^{-3}, \ldots, 10^3\}$. Thus, 49 combinations of $C$ and $\sigma^2$ were tried in this step. An optimal pair $(C_0, \sigma^2_0)$ was selected from this coarse grid search. In the second step, a fine grid search was conducted around $(C_0, \sigma^2_0)$. The complete multiclass SVM-RFE process is described as follows:

1. Start with an empty ranked features list $R = []$ and the selected feature list $F = [1, \ldots, d]$.
2. Repeat until all features are ranked:
   - (a) train $n(n-1)/2$ or $n-1$ fuzzy SVMs with all the training samples, with all features in $F$;
   - (b) compute and sum the ranking criterion $c_{st}$ of $n(n-1)/2$ or $n-1$ SVMs for features in $F$ using Eq. (16);
   - (c) find the feature with the smallest ranking criterion: $e = \arg \min_y c_{ef}$;
   - (d) add the feature $e$ into the ranked feature list $R$: $R = [e, R]$;
   - (e) remove the feature $e$ from the selected feature list $F$: $F = F - [e]$.
3. Output: Ranked feature list $R$.

4. Experimental design

A total of 69 mobile phones were collected from the Taiwan marketplace in 2006. Three product designers, each with at least 5 years experience, conducted the product form features analysis. They first examined the main component structures using the method proposed by Kwak and Han (2002) and then used it to analyze all the product samples. The form features of each product sample were discussed by all three designers, who then determined one unified representation. Twelve form features of the mobile phones’ designs, including four continuous attributes and eight discrete attributes, were used in this study. A complete list of all product form features is shown in Table 1. Notice that the color and texture information of the product samples was ignored and that emphasis was placed on the form features only. All entities in the feature matrix were prepared for training in the multiclass fuzzy SVM model. Five class labels such as sports, simplicity, female, plain and business were chosen for semantic evaluations. In order to collect consumers’ perception data of the mobile phones’ designs, 30 subjects, 15 men and 15 women, were asked to evaluate all the product samples using the five selected class labels. Each subject was asked to choose the most suitable class label to represent each product sample, and evaluate each sample on a semantic differential scale from 0 (very low) to 1 (very high). Since there was only a single instance of each product sample when training the multiclass fuzzy SVM model, the ones with the most frequently assigned label were used for representing each product sample. The training multiple instances of samples for SVM is another interesting issue worthy of further research. The selected class label was designated as $+1$, and the rest of the labels were designated as $-1$. The semantic differential score was directly stored as the membership value for fuzzy SVM training.
Table 1
Complete list of product form features of mobile phone design

<table>
<thead>
<tr>
<th>Form features</th>
<th>Type</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>Continuous</td>
<td>None</td>
</tr>
<tr>
<td>Width</td>
<td>Continuous</td>
<td>None</td>
</tr>
<tr>
<td>Thickness</td>
<td>Continuous</td>
<td>None</td>
</tr>
<tr>
<td>Volume</td>
<td>Continuous</td>
<td>None</td>
</tr>
<tr>
<td>Type</td>
<td>Discrete</td>
<td>Block body</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{51})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Flip body</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{52})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Slide body</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{53})</td>
</tr>
<tr>
<td>Type</td>
<td>Discrete</td>
<td>Full</td>
</tr>
<tr>
<td></td>
<td></td>
<td>separated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{61})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Partial</td>
</tr>
<tr>
<td></td>
<td></td>
<td>separated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{62})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Regular</td>
</tr>
<tr>
<td></td>
<td></td>
<td>separated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{63})</td>
</tr>
<tr>
<td>Style</td>
<td>Discrete</td>
<td>Round</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{71})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Square</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{72})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bar</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{73})</td>
</tr>
<tr>
<td>Shape</td>
<td>Discrete</td>
<td>Circular</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{81})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Regular</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{82})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Asymmetric</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{83})</td>
</tr>
<tr>
<td>Arrangement</td>
<td>Discrete</td>
<td>Square</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{91})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vertical</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{92})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Horizontal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{93})</td>
</tr>
<tr>
<td>Detail treatment</td>
<td>Discrete</td>
<td>Regular</td>
</tr>
<tr>
<td></td>
<td></td>
<td>seam</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{101})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Seamless</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{102})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Vertical</td>
</tr>
<tr>
<td></td>
<td></td>
<td>seam</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{103})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Horizontal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>seam</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{104})</td>
</tr>
<tr>
<td>Position</td>
<td>Discrete</td>
<td>Middle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{111})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Upper</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{112})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lower</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{113})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Full</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{114})</td>
</tr>
<tr>
<td>Panel</td>
<td>Discrete</td>
<td>Square</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{121})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fillet</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{122})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Shield</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{123})</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Round</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(X_{124})</td>
</tr>
</tbody>
</table>
5. Experimental results and analyses

5.1. Determination of optimal training model

The results of cross-validation for the Gaussian kernel model are shown in Fig. 1. As shown in Fig. 1(a), the best parameter set \((C_0, \sigma^2_0)\), obtained from the first step of the coarse grid search, was \((10, 10)\); determined by choosing the lowest error rate of 73.9%. As shown in Fig. 1(b), a fine grid search was conducted around \((10, 10)\) in the second step. The optimal parameter set \((C, \sigma^2)\), obtained from a fine grid search, was \((40, 4)\); determined by choosing the lowest error rate of 72.4%. The training error rate also improved slightly from the first step to the second step. If the training model is built with whole data samples without cross-validation and selection of only one of the parameter sets from regions with very low average error rates, the training model can barely handle the overfitting problem. The process of cross-validation is capable of balancing the trade off between improved training accuracy and preventing overfitting. The best parameter set \((C, \sigma^2) = (40, 4)\) of the Gaussian kernel obtained from the cross-validation process was then used to build the final multiclass fuzzy SVM training model. The average accuracy rate of the optimal Gaussian kernel model was 98.6%. Further analysis using the confusion matrix in Table 2 shows that this model performed very well and had only one misclassified sample ("plain" misclassified to "female"), as shown in Table 2.

5.2. Analysis of feature ranking

Table 3 shows the results of overall ranking and class-specified ranking. For the overall ranking – by considering all five class labels – the feature \(X_{10}\) (style of number button) was the most important. This result was consistent with the class-specific ranking of \(Y_2\) (sports), \(Y_3\) (female), and \(Y_4\) (simplicity). However, for class-specific ranking \(Y_1\) (plain), and \(Y_5\) (business), the most important form feature was \(X_6\) (arrangement of function button), and \(X_7\) (style of function joystick), respectively. For the last five features, overall ranking and class-specific ranking were both the same as \(X_1, X_2, X_3, X_4, X_9\), but the order of ranking was different for different class labels. The least important feature of both the overall ranking and the class-specific ranking was \(X_9\) (arrangement of number button).

5.3. Performance of selected feature subset

Fig. 2 shows the performance of the feature subset of the overall ranking and the class-specified ranking. As described in the previous section, the five least important features in the overall ranking were \(X_2 > X_4 > X_3 > X_1 > X_9\). The performance of the feature subset of the overall ranking shown in Fig. 2(a) provided more information than just the feature ranking. The average error rates of the first five steps were very low at less than 1%. This indicates that the remaining seven features can be used to build a classification model with very high predictability. However, the average error rates increased more drastically if more than seven features were eliminated. Product designers can select the smallest feature subset with a given discrimination capability. For example, if the allowable average error rate for the overall ranking is 30%, the last nine features can be eliminated. Designers only need to concentrate on the top three features for the product form design. If designers specifically want to consider more for each class of label, the performance of a feature subset using class-specific ranking shown in Fig. 2(b) can also be used in the same manner. Notice that the average error rates of all class labels were very low: within 10% in the first
six steps. When the 7th and 8th features were eliminated, the error rate of label $Y_1$ was larger than the other four labels. For example, take the same allowable average error rate of 30%: nine features can be eliminated for label $Y_4$ while only eight features can be eliminated for labels $Y_1$, $Y_2$, $Y_3$, $Y_5$.

5.4. Weight distribution analysis of overall ranking

Fig. 3 shows the weight distribution of overall ranking by summing the criterion $c_{st}$ of $n(n - 1)/2$ OVO SVMs. Fig. 3(a) shows the weight distribution of Step 1 before removing any feature. The feature $X_9$ with the lowest weight value was removed in this step because the arrangement of the number buttons for all the product samples was the same, thus this feature had the least influence on the classification results. The features with more weight (e.g. $X_9$, $X_5$) can be regarded as being strongly correlated with the class distinction, that is, these features had much influences to distinguish samples belong to which class. The feature $X_5$ (body type) had the highest weight equal to 1.0. This indicates that the classification of the product sample is strongly related to the three kinds of body type.
including block, flip and slide body. In contrast, the features \( X_2 > X_3 > X_4 > X_1 \) with less weight had less influence, and, therefore, were eliminated during the next stage (steps 2 ~ 5) of the multiclass SVM-RFE process. The features \( X_1, X_2, X_3 \) and \( X_4 \), with less weight, denote the length, width, thickness and volume of the object, respectively. This result makes sense since most mobile phones today are slim and light-weight, and there is not much difference in their size and shape. Fig. 3(b) shows the weight distribution of step 6 after eliminating features \( X_1, X_2, X_3 \) and \( X_4 \). It was very interesting to find that the weight of \( X_5 \) was larger than \( X_{10} \) in the first six steps. However, the weight of \( X_{10} \) became larger than \( X_5 \) in Step 7 as shown in Fig. 3(c). This indicates that the style of the number button in this step became more important than the body type when eliminating some less important features. As shown in Fig. 3(d), the difference in weight values between \( X_{10} \) and the remaining features \( X_{10} > X_5 > X_7 \) became largest in Step 9.

5.5. Weight distribution analysis of class-specific ranking

Fig. 4 shows the weight distribution of the class-specific ranking of the different labels. The weight distribution before removing any features was shown in Fig. 4(a). The feature with the largest weight was feature \( X_5 \) for label \( Y_1 \), \( Y_2 \), \( Y_4 \), \( Y_5 \) and \( X_{10} \) for label \( Y_3 \). In the first step, those features with the largest initial weight had the largest influence on the classification results. However, they were not necessarily the most important features in the final ranking. As can be seen from the results, the most important features obtained in Table 3 were \( X_6 \) for label \( Y_1 \), \( X_{10} \) for label \( Y_2 \), \( Y_3 \), \( Y_4 \), and \( X_7 \) for \( Y_5 \). As for the less important features, for example, the five least important features \( X_1, X_2, X_3, X_4, X_9 \) were all the same for all labels. Those features with a smaller weight were all eliminated in the following steps. Fig. 4(b) shows the weight distribution after eliminating the five least important features. As shown in Fig. 4(c), the features with the largest weight in
Step 8 became $X_5$ for label $Y_1$ and $X_{10}$ for labels $Y_2$, $Y_3$, $Y_4$, $Y_5$.

6. Conclusions

Selecting critical product form features according to consumer preference is very useful for product designers, yet only a few attempts have been made to do this in the product design field. In this paper, an approach based on multiclass SVM-RFE is proposed to identify and analyze important product form features. The fuzzy SVM model can deal with the nonlinear relationship of product form features by using the Gaussian kernel function. The optimal training parameters can be determined by a two-step cross-validation process. In our case study of mobile phone design, the optimal Gaussian kernel model was obtained by choosing the lowest average error rate (72.4%) of cross-validation. The parameter set $(C, \sigma^2)$ of the optimal training model was (40, 4). The optimal Gaussian kernel training model also had a very high accuracy rate of 98.6% with all product samples. Finally, the multiclass SVM-RFE process based on this optimal Gaussian kernel model was used to analyze the form features of product samples. Either by considering the overall ranking or the class-specific ranking, the less important form features can be eliminated and still provide a very high classification performance. The multiclass SVM-RFE process has proved to be very useful in finding the subset of form features with minimum generalization and in selecting the smallest possible subset with a given discrimination capability.

References


