A Selective Multiclass Support Vector Machine Ensemble Classifier for Engineering Surface Classification Using High Definition Metrology

Shichang Du[1]
State Key Lab of Mechanical
System and Vibration,
School of Mechanical Engineering,
Shanghai Jiao Tong University,
Shanghai 200240, China
e-mail: lovbin@sjtu.edu.cn

Changping Liu
School of Mechanical Engineering,
Shanghai Jiao Tong University,
Shanghai 200240, China
e-mail: lcpsky89@gmail.com

Lifeng Xi
State Key Lab of Mechanical
System and Vibration,
School of Mechanical Engineering,
Shanghai Jiao Tong University,
Shanghai 200240, China
e-mail: lfxi@sjtu.edu.cn

The surface appearance is sensitive to change in the manufacturing process and is one of the most important product quality characteristics. The classification of workpiece surface patterns is critical for quality control, because it can provide feedback on the manufacturing process. In this study, a novel classification approach for engineering surfaces is proposed by combining dual-tree complex wavelet transform (DT-CWT) and selective ensemble classifiers called modified matching pursuit optimization with multiclass support vector machines ensemble (MPO-SVME), which adopts support vector machine (SVM) as basic classifiers. The dual-tree wavelet transform is used to decompose three-dimensional (3D) workpiece surfaces, and the features of workpiece surface are extracted from wavelet sub-bands of each level. Then MPO-SVME is developed to classify different workpiece surfaces based on the extracted features and the performance of the proposed approach is evaluated by computing its classification accuracy. The performance of MPO-SVME is validated in case study, and the results demonstrate that MPO-SVME can increase the classification accuracy with only a handful of selected classifiers.

[DOI: 10.1115/1.4028165]

Keywords: surface classification, dual-tree complex wavelet transform, feature extraction, support vector machine, selective ensemble

1 Introduction

The condition of a workpiece surface has great impact on the functional performance of the product, and the classification of the workpiece surface patterns plays an important role in quality control due to its function in providing feedback on the manufacturing process [1].Since it is not appropriate to directly use the raw data collected from the workpiece surfaces to classify, features need to be extracted first to represent a given workpiece surface before classification. Conventionally, some numerical surface parameters are used for surface-texture characterization, such as amplitude parameters like average roughness, maximum profile peak height, spacing parameters like mean peak spacing and high spot count, and shape parameters like slope and kurtosis [2–4]. But with the development of 3D measuring techniques especially faster optimal methods, a lot of data can be obtained from engineering surfaces in quite a short time [5,6], it is inappropriate to simply use one-dimensional (1D) parameters to characterize surface textures. Consequently, more surface characterization approaches have been proposed to obtain abundant information about 3D surfaces, such as a 3D parameter set [4,7–9], gray level co-occurrence matrix [10], two-dimensional (2D) autocorrelation function and spectral analysis [11], and a 3D Monte Carlo model [12]. In recent years, some approaches that first used in signal processing areas have been adopted to extract features of engineering surfaces, such as Gabor filter banks [13], Gaussian filter banks [14], and wavelet packets [15–17], and after filtering, some numerical surface parameters are calculated for each sub-band to represent a given surface.

The goal of feature extraction is to improve the effectiveness and efficiency of classification. Namely, the performance of feature extraction method can affect the accuracy of classification directly. So it is of great importance to select an appropriate approach to extract features. In this study, attention is mainly focused on wavelet filters, due to the fact that it can provide multiscale/orientation analysis, which makes it a powerful tool in feature extraction and is superior to traditional filters [16,18]. There are many kinds of wavelets, and discrete wavelet transform (DWT) is the most widely used one in the analysis of workpiece surfaces in previous studies [16,19,20]. However, DWT mainly has two disadvantages: lack of shift-invariance and poor directional selectivity for diagonal features, which impair its application in engineering surface analysis. This paper adopts DT-CWT, which is a type of wavelet filter proposed by Kingsbury [21–24] in recent years, to extract features of engineering surfaces. It turns out that both of the above two problems can be solved effectively by DT-CWT; furthermore, DT-CWT also has some other nice properties as follows: (1) perfect reconstruction, (2) limited redundancy, (3) efficient order-N computation. Due to the above merits, DT-CWT performs well in feature extraction.

SVM, proposed by Vapnik [25], is a new machine learning method developed on the basis of statistical learning theory. It can solve a variety of problems such as classification, regression, and prediction [26–29]. SVM produces good generalization abilities by constructing a best hyperplane as the decision surface to maximize the margin between the data of two classes. For nonlinear classification problems, SVM uses a kernel function to map the original dataset into a high dimensional space, in which the data is linear and separable, so it overcomes the “dimension disaster” problem, which faced by many traditional classification methods.

Classifier combination is now an active area of research in machine learning and pattern recognition. Many studies, both theoretically and empirically, demonstrate that combining classifiers

[1]Corresponding author.

Contributed by the Manufacturing Engineering Division of ASME for publication in the Journal of Manufacturing Science and Engineering. Manuscript received June 3, 2013; final manuscript received July 30, 2014; published online November 26, 2014. Assoc. Editor: Robert Gao.

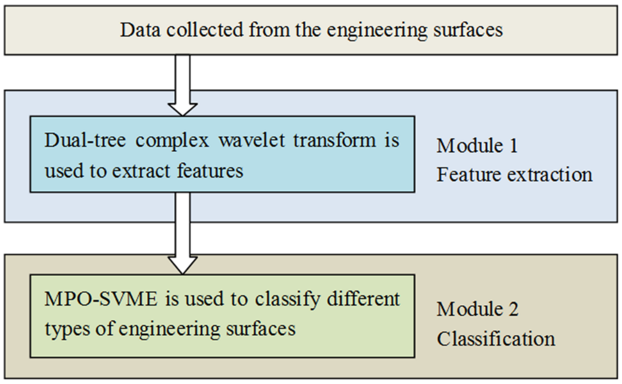

Data collected from the engineering surfaces
Dual-tree complex wavelet transform is used to extract features
Module 1
Feature extraction
MPO-SVME is used to classify different types of engineering surfaces
Module 2
Classification

Fig. 1 The architecture of the proposed feature extraction and classification approach

instead of using a single one can improve generalization performance of SVMs and increase the classification accuracy [30–33]. An ensemble of classifiers consists of several basic classifiers created by using several strategies on the training data and aggregates their outputs to classify the testing samples. Classifiers ensemble can deal with some problems that are intractable for a single classifier due to its higher generalization abilities compared with a single one. Though it is well known that SVM ensembles have been applied in many different areas, e.g., Face membership authentication [34], text-independent speaker recognition [35], bankruptcy prediction [36], process mean shifts classification [29], few researches have been done to apply SVMs ensemble technique into classifying the engineering surfaces in the manufacturing areas.

Therefore, this paper developed a novel method called modified matching pursuit optimization with multiclass support vector machines ensemble (MPO-SVME) to solve the classification problem of engineering surfaces. MPO-SVME is a selective ensemble multiclass SVMs, which can improve the classification performance by creating an ensemble consist of several multiclass SVMs and selecting the optimal ones from the ensemble. The significance of the present method is that it can improve the classification accuracy without bringing in too much computational cost, which means that it can be applied to a practical manufacturing system.

The remainder of this paper is organized as follows. Section 2 first reviews the DT-CWT theory and demonstrates how to use it to extract features, then describes the SVMs theory and the details of the proposed MPO-SVME algorithm. In Sec. 3, a case study is presented to demonstrate the effectiveness of using DT-CWT to extract features and using MPO-SVME to classify real-world engineering surface data. Also in this section, the performance analysis is implemented to illustrate how the parameters affect the classification performance for high resolution surface samples. Finally, Sec. 4 draws conclusions of this work.

2 The Proposed Methodology

This section provides an overview of the developed method which consists of DT-CWT and selective ensemble SVMs, and the architecture is shown in Fig. 1. To be specific, in Module 1, high resolution images of engineering surfaces are filtered to extract details in different directions and scales using 2D DT-CWT, and then the means and standard variances of the coefficients of the wavelet sub-bands are calculated as feature vectors to represent a given surface. In Module 2, random subspace algorithm is first used to create several component multiclass SVM classifiers and then MPO algorithm is conducted to select some optimal classifiers from the available ones.

2.1 Feature Extraction With DT-CWT

2.1.1 The DT-CWT. The DT-CWT is a multiscale transform tool that uses a dual tree of wavelets filters to obtain the real and imaginary parts of complex wavelets coefficients. It not only retains the merits of traditional wavelet transform but also provides some nice properties such as approximate shift invariance and improved directional selectivity. The 1D DT-CWT is implemented using two filter banks in parallel to operate on the input data $f(x)$, and it is composed of two parallel fully decimated trees of filters, tree a and tree b. Tree a produces the real parts of the complex coefficients while tree b produces the imaginary parts. A 4-level decomposition implemented by 1D DT-CWT is illustrated in Fig. 2, where $h_0(n)$ and $h_1(n)$ are the low-pass and high-pass filters of tree a, and g_0 (n) and $g_1(n)$ are the low-pass and high-pass filters of tree b.

The high-dimensional DT-CWT not only maintains all the attractive properties of the 1D DT-CWT but also provides directional selectivity, which makes it more suitable to extract features from engineering surfaces. Figure 3(a) shows the 2D discrete wavelet filters, note that the first two wavelets are oriented in only three directions (vertical, horizontal, and diagonal). Figure 3(b) shows the 2D dual-tree complex wavelet filters, where the first row illustrates the real part of each complex wavelet, the second row illustrates the imaginary part, and the third row illustrates the magnitude. Note that the 2D dual-tree complex wavelet filters are oriented in six directions (±15 deg, ±45 deg, and ±75 deg), so it

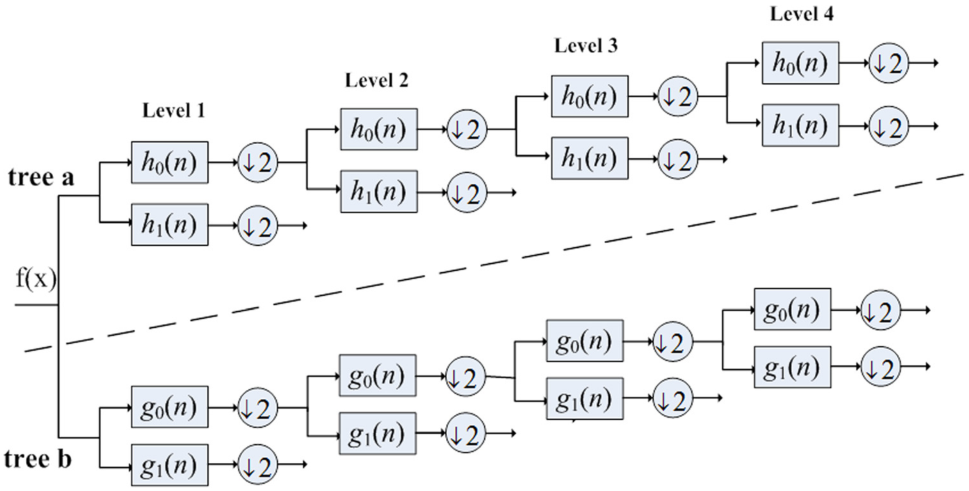
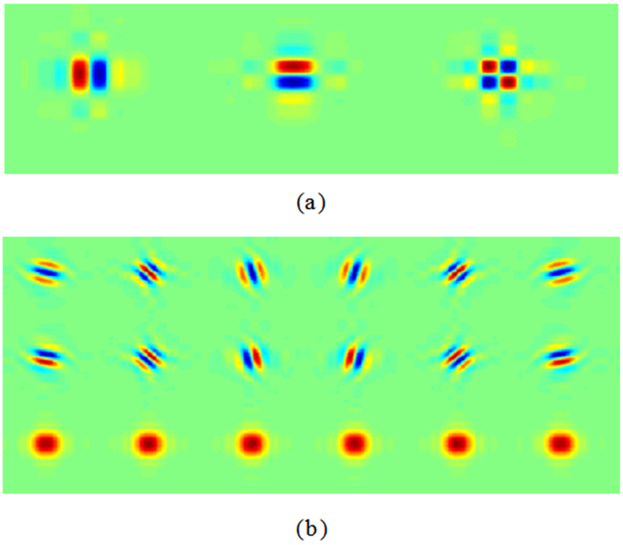


Fig. 2 Framework of the DT-CWT


(a)

(b)

Fig. 3 The two set of wavelet filters (*a*) 2D discrete wavelet filters and (*b*) 2D dual-tree complex wavelet filters

does better in reflecting the change of images in different orientations.

2.1.2 Feature Extraction. Feature extraction is related to the quantification of surface characteristics, and its quantitative results are known as feature vectors. The selection of these descriptive parameters is important, because it will influence the subsequent classification performance to some degree. A K-level 2D DT-CWT decomposition is implemented to extract features. Near-symmetric 13, 19 tap filters are selected as level 1 filters and Q-shift 18 tap filters for other levels, which has been proved by Kingsbury [23] to be the best combination of filters when compared with other combinations of DT-CWT filters. Feature vectors are generated by computing means and standard deviations from coefficients of wavelet sub-bands [37]

$$\mu = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}|C(i,j)| \tag{1}$$

$$\sigma = \sqrt{\frac{1}{M \times N - 1}\sum_{i=1}^{M}\sum_{j=1}^{N}|C(i,j) - \mu|^2} \tag{2}$$

As shown in Eqs. (1) and (2), M denotes the number of points along x axis, N denotes the number of points along y axis, $C(i,j)$ denotes the coefficient matrix of wavelet sub-bands, μ and σ, respectively, denote the means and standard deviations of the sub-bands coefficients.

Next, by computing means and standard deviations from coefficients of wavelet sub-bands of low-pass in K levels, $2 \times K$ characteristic values are obtained, composing a $2 \times K$ dimensional feature vector $(\mu_1, \mu_2, ..., \mu_K, \sigma_1, \sigma_2, ..., \sigma_K)$. To wavelet sub-bands of high-pass, means and standard deviations from coefficients of each direction (±15 deg, ±45 deg, and ±75 deg) in each of K levels are calculated and a $12 \times K$ dimensional feature vector is generated $(\mu_{11}, \mu_{12}, ..., \mu_{K5}, \mu_{K6}, \sigma_{11}, \sigma_{12}, ..., \sigma_{K5}, \sigma_{K6})$. Through combining the $12 \times K$ dimensional high-pass feature vector and $2 \times K$ dimensional low-pass feature vector, a $14 \times K$ dimensional feature vector is obtained to represent a given surface.

2.2 SVMs

2.2.1 Brief Review of SVMs. The basis of SVMs is to perform classification by transforming the data into a high dimensional feature space to find the optimal hyperplane that maximizes the margin between the two classes. The vectors that define the hyperplane are called the support vectors. Given a training data set of instance-label pairs $\{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), ...(\mathbf{x_i}, y_i), ...(\mathbf{x_N}, y_N)\}$, $i = 1, 2, \cdots, N$, where N is the total number of training samples, $\mathbf{x_i} \in R^d \subset R$ is the ith d-dimensional input vector, and $y_i \in \{1, -1\}$ is the known target.

The training of SVM requires solving the following optimization problem:

$$\text{Minimize } \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i \tag{3}$$

$$\text{Subject to } y_i(\mathbf{w}^T\phi(\mathbf{x_i}) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{4}$$

Where ξ_i denotes slack variables, measuring the degree of misclassification of the sample $\mathbf{x_i}$, C is a penalty parameter, which is used to penalize training errors. The bias b is a scalar, representing the bias of the hyperplane, $\mathbf{w}$ is the vector of hyperplane coefficients, defining a direction perpendicular to the hyperplane, the index i labels the N training cases, and the map function ϕ is a nonlinear transformation used to map the input vectors into a high dimensional feature space (as shown in Fig. 4). The optimization problem is to determine the trade-off between margin size and training error.

For linearly separable problems, $\xi_i = 0$ and the separating hyperplane that creates the maximum distance between the plane and the nearest data is the optimal one. For the nonlinear classification tasks, a mapping function is often employed to map the training samples from the input space into a higher dimensional feature space, and then the nonlinear classification problem will become a linear one. Any function that satisfies Mercer's theorem [27] can be used as a kernel function. Though new kernels are being proposed by researchers, the most widely used kernels are as follows:

Linear function: $K(\mathbf{x}_i, \mathbf{y}_j) = \mathbf{x}_i \times \mathbf{y}_j$

Gaussian radial basis function (RBF): $K(\mathbf{x}_i, \mathbf{y}_j)$

$= \exp(-\gamma\|\mathbf{x}_i - \mathbf{y}_j\|^2)\gamma > 0$

Polynomial function with degree d: $K(\mathbf{x}_i, \mathbf{y}_j) = ((\mathbf{x}_i \times \mathbf{y}_j) + \gamma)^d\gamma > 0$

Sigmoid: $K(\mathbf{x}_i, \mathbf{y}_j) = \tan h(\gamma\mathbf{x}_i \times \mathbf{y}_j + r)\gamma > 0$

2.2.2 Multiclass SVMs. SVM is in nature a tool for binary classification [25], which is not suitable for classifying

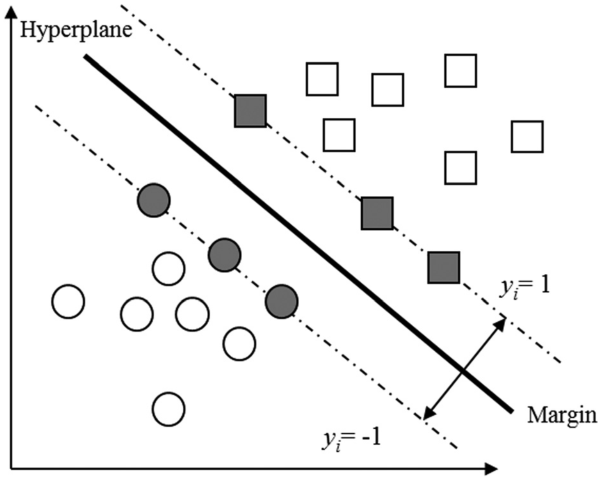


Fig. 4 A geometric interpretation of binary classification of SVM

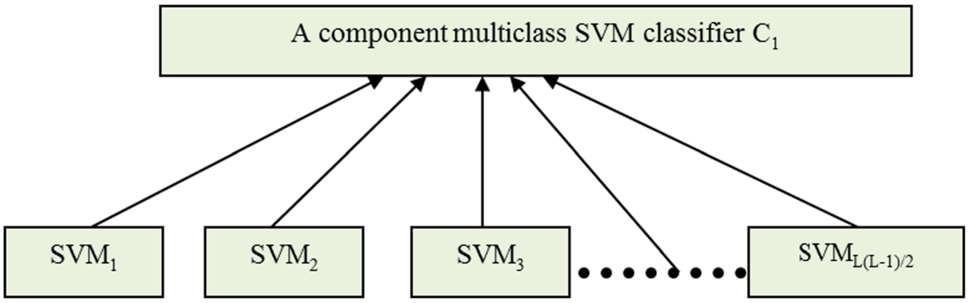


Fig. 5 A component multiclass SVM classifier combined by binary SVMs

engineering surfaces because there are usually several different cases needed to be identified. Thus, it is necessary to extend the binary classifiers in order to solve multiclass problems. The traditional way to implement multiclass classification is to combine several binary SVMs; in light of this, three popular methods, "one-against-one," "one-against-all," and direct acyclic graph (DAG), are proposed to solve multiclass problems. A comprehensive comparison of these three methods conducted by Hsu and Lin [38] suggested that the one-against-one method is most suited for practical use than other methods. Therefore, this study adopts one-against-one method to classify workpiece surface patterns.

Figure 5 shows how to construct a component multiclass SVM classifier using one-against-one strategy. For L-class event, the one-against-one method constructs $M = C_L^2 = L(L-1)/2$ binary classifiers, each of which is trained by binary-class data. For training data from the ith and the jth class, the following binary classification problem needs to be solved:

$$\min \frac{1}{2}(\mathbf{w}^{ij})^T \mathbf{w}^{ij} + C \sum_t \xi_n^{ij} (\mathbf{w}^{ij})^T$$

$$\text{s.t.} \begin{cases} (\mathbf{w}^{ij})^T K(\mathbf{x_n}) + b^{ij} \geq 1 - \xi_n^{ij} & \text{if} \quad y_n = i \\ (\mathbf{w}^{ij})^T K(\mathbf{x_n}) + b^{ij} \leq \xi_n^{ij} - 1 & \text{if} \quad y_n = j \\ \xi_n^{ij} \geq 0 \end{cases} \qquad (5)$$

where similarly to binary classification SVMs, $K(\mathbf{x_n})$ is the kernel function, $(\mathbf{x_n}, y_n)$ is the ith or jth training sample, $\mathbf{w} \in R^N$ and $b \in R$ are the weighting factors, ξ_n^{ij} is the slack variable, and C is the penalty parameter.

Next, voting strategy is used to output the results after all the $L(L-1)/2$ classifiers are constructed: If $((\mathbf{w}^{ij})^T K(\mathbf{x_n}) + (b^{ij}))$ predicts $\mathbf{x_n}$ is in the ith class, the vote for the ith class increased by one, otherwise, the jth class added by one. Then $\mathbf{x_n}$ is finally predicted in the class with the largest vote. The voting approach described above is also called the "Max Wins" strategy.

2.3 The Proposed MPO-SVME Classifier

2.3.1 Framework of MPO-SVME. The main steps of MPO-SVME are shown in Fig. 6.

Step 1: Creation of several training subsets using ensemble strategies. In this paper, random subspace algorithm (discussed in Sec. 2.3.2) is adopted to create several different training subsets S$_1$, S$_2$,…, S$_T$.

Step 2: Creation of several component multiclass SVM classifiers using one-against-one method (discussed in Sec. 2.2.2). Based on training subsets created in step 1, several multiclass SVM classifiers are constructed and used as component multiclass SVMs classifiers for optimal selection in step 3, and they are denoted by $C_1, C_2, \cdots C_T$.

Step 3: Selection of optimal classifiers. Instead of trying to ensemble all the independent component multiclass SVM classifiers created in step 2, the optimal subset is selected from the component multiclass SVM classifiers through MPO algorithm (discussed in Sec. 2.3.3). And the final prediction of the selective

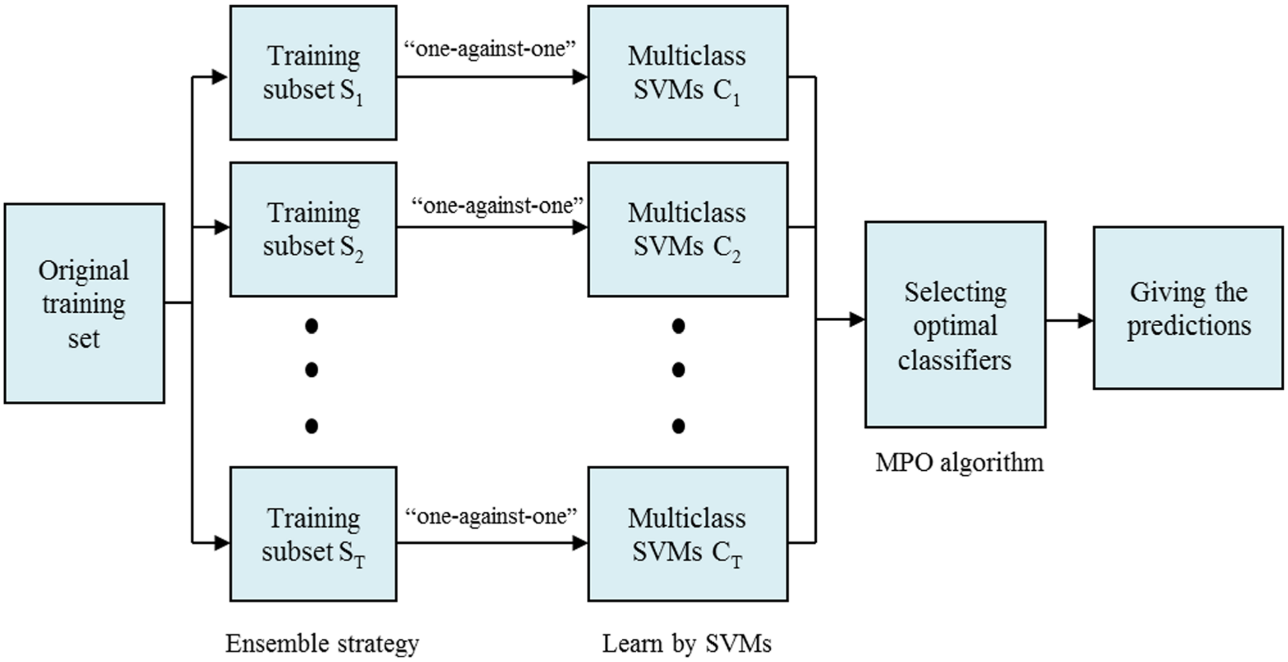


Fig. 6 The framework of the proposed selective ensemble classifiers

ensemble classifiers is given by the aggregation of the predictions of selected component classifiers while taking their weights into consideration.

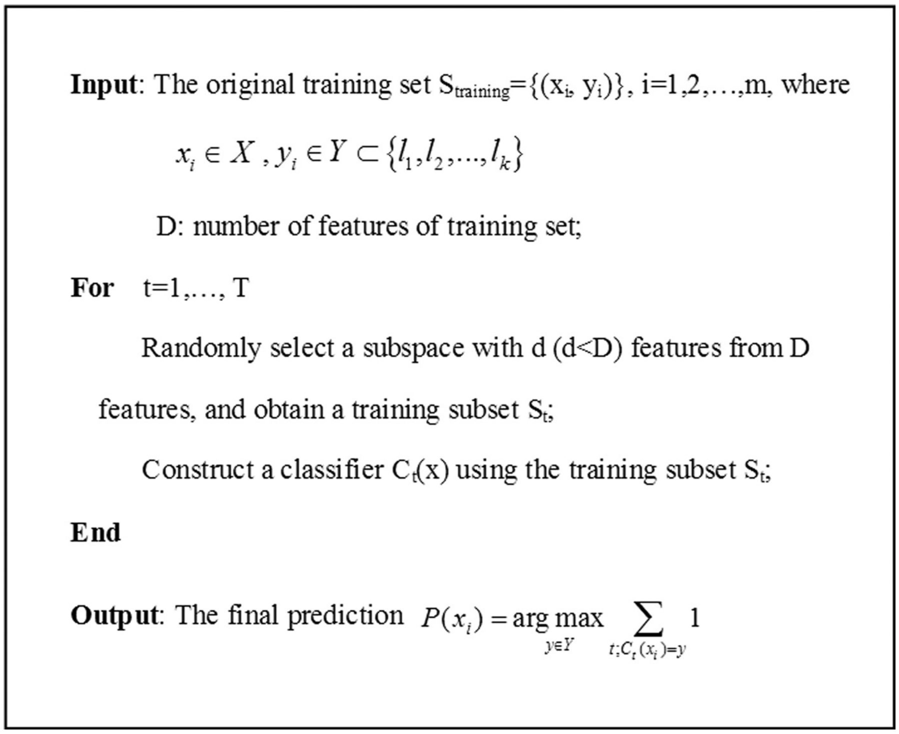
2.3.2 Ensemble Strategy. An ensemble of classifiers is a collection of several classifiers (usually trained on different training subsets) whose individual decisions are combined in some way to classify the test samples. It is known that an ensemble often shows better performance than the individual classifiers that make it up. Therefore, many approaches of constructing SVM ensembles have been proposed, such as bagging [39], boosting [40], and random subspace [41]. In bagging, some new training sets are generated from the original training set via a bootstrap method and then used to train several corresponding classifiers. The main idea behind boosting is to construct a composite classifier by training classifiers sequentially while putting more and more emphasis on misclassified cases. In this study, random subspace is adopted to ensemble learning.

Similar to bagging, the random subspace algorithm also benefits from bootstrapping. However, other than bootstrapping training samples as used in bagging, random subspace performs the bootstrapping in the feature space. It is well known that over-fitting is likely to happen when the size of training set is small while the dimensionality of the feature vector is relatively high, and this problem is pretty common in texture or surface classification due to the fact that the size of feature vector is often larger than the numbers of training samples. In order to avoid over-fitting, random subspace method randomly selects a small subset of features to reduce the discrepancy between the training data size and the feature vector length. Using a random sampling method, we construct a set of SVMs. Then these SVMs are combined to construct a more powerful classifier to solve the over-fitting problem. The random subspace algorithm is shown in Fig. 7.

2.3.3 Selective Ensemble Strategy. Zhou et al. [42] proposed a concept which has been proved theoretically and empirically in his article that selecting some suitable classifiers from the available classifiers may yield better accuracy than combing all classifiers simultaneously, and this opinion can be deemed as "Many could be better than all." In this paper, "ensemble all" means ensemble the output results of all the component multiclass SVM classifiers created by random subspace algorithm. The value of selective ensemble lies in reducing the size of ensemble without worsening its performance or even getting a better result. However, it is really a tough task to exclude the underperformed classifiers and retain the optimal subsets since the subspace of possible subsets is very large ($2^T - 1$) for a size of T (T is the total number of component SVM classifiers). Therefore, it is not practical to use the exhaustive method to search for optimal subset. A large number of very diverse selective ensemble methods have been proposed over the past decade, such as search-based methods like Gasen [42] and ensemble pruning via reinforcement learning (EPRL) [43], clustering-based methods like hierarchical agglomerative clustering (HAC) [44], and ranking-based method like "Pruning in Ordered Bagging Ensembles" [45]. So far, there is still no agreement about which is the most appropriate method. In this paper, we adopt a novel selective ensemble approach named matching pursuit optimization ensemble classifiers (MPOEC), which is proposed by Mao et al. [46] and has been proven to have better performance than some existing methods.

The main idea of MPOEC is to determine the ideal balance between the diversity and the individual accuracy of the ensembles. MPOEC adopts a greedy iterative algorithm of matching pursuit to search for an optimal combination of entire classifiers, and abandons some similar or poor classifiers by giving zero coefficients. In MPOEC, each classifier of the ensemble is deemed as a basis function and the labels of training samples as the target function. The coefficient of each classifier is obtained by minimizing the residual between the target function and the linear combination of individual component classifiers.

The set of all component classifiers $\{C_t(x)\}(t = 1, ..., T)$ is regarded as the basis function dictionary, and every component classifier can obtain a coefficient α_t by minimizing the residual R. When $\alpha_t \neq 0$, the classifier corresponding to α_t is selected to ensemble, and when $\alpha_t = 0$, the classifier will be excluded. However, it should be noted that the MPOEC algorithm in Ref. [46] can only solve the 2-class classification problem, which is not suitable for classification of various workpiece surfaces. So we



```
Input: The original training set $S_{training}=\{(x_i, y_i)\}$, i=1,2,…,m, where
        $x_i \in X$, $y_i \in Y \subset \{l_1, l_2, ..., l_k\}$
      D: number of features of training set;
For   t=1,…, T
      Randomly select a subspace with d (d<D) features from D
   features, and obtain a training subset $S_t$;
      Construct a classifier $C_t(x)$ using the training subset $S_t$;
End
Output: The final prediction $P(x_i) = \arg\max_{y \in Y} \sum_{t:C_t(x_i)=y} 1$
```

Fig. 7 The random subspace algorithm

modify this algorithm to solve the multiclass classification problem, and since SVM is adopted as basic classifiers in this study, the modified algorithm is then called MPO-SVME. The description of MPO-SVME algorithm is shown in Fig. 8.

3 Case Study

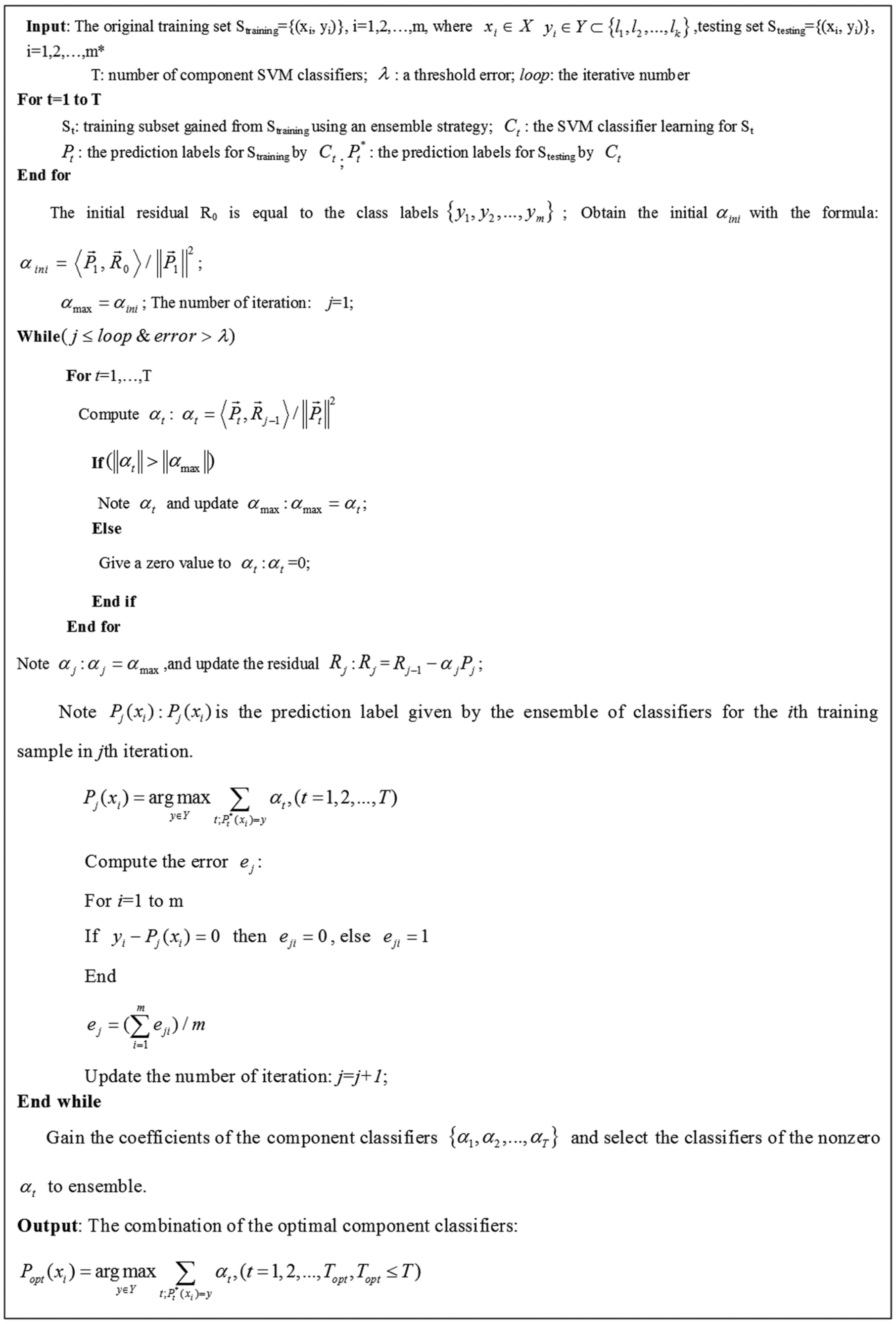
Engineering surfaces, which are represented by point cloud data collected by a device called Coherix ShaPix [47], are used to validate the effectiveness of the proposed method on real-world data. ShaPix uses a tunable-wavelength laser to gather up to 1×10^6 data points about the part in its 300 mm by 300 mm field of view in less than a minute. The basic height (Z) accuracy of it is 1 μm while the lateral (X, Y) resolution is around 0.3 mm.

Six different top surfaces of engine cylinder blocks (as shown in Fig. 9) processed by a major domestic car manufacturer are studied here, and it is worth noting that engineering surfaces 1–3 are finished by one machine while surfaces 4–6 by another. Coherix ShaPix is used to scan these six surfaces and gather more than 700,000 data (three-co-ordinate data) from each surface. Two samples of the color-coded measurement results from each data set are shown in Fig. 10.

It can be seen that parts in set A have a slope problem (the height decreases from bottom left to upper right) while parts in set B don't, and this problem will lead to some adverse effects on the performance of the product. So it is worthwhile to distinguish these problematic parts from the normal ones by classification.

3.1 Feature Extraction.

Twenty square areas are designated from the entire surface with the size of 128 × 128 pixels each. Centroids of those square areas are located in the intersections of the grids, as shown in Fig. 11. One of the square areas rather than



```
Input: The original training set $S_{training}=\{(x_i, y_i)\}$, i=1,2,…,m, where $x_i \in X$ $y_i \in Y \subset \{l_1, l_2, ..., l_k\}$ ,testing set $S_{testing}=\{(x_i, y_i)\}$,
i=1,2,…,m*
        T: number of component SVM classifiers; $\lambda$ : a threshold error; loop: the iterative number
For t=1 to T
    $S_t$: training subset gained from $S_{training}$ using an ensemble strategy; $C_t$ : the SVM classifier learning for $S_t$
    $P_t$ : the prediction labels for $S_{training}$ by $C_t$ ; $P_t^*$ : the prediction labels for $S_{testing}$ by $C_t$
End for
    The initial residual $R_0$ is equal to the class labels $\{y_1, y_2, ..., y_m\}$ ; Obtain the initial $\alpha_{ini}$ with the formula:
$\alpha_{ini} = \langle \vec{P}_1, \vec{R}_0 \rangle / \|\vec{P}_1\|^2$ ;
    $\alpha_{max} = \alpha_{ini}$ ; The number of iteration: j=1;
While($j \le loop \,\&\, error > \lambda$)
    For t=1,…,T
        Compute $\alpha_t$ : $\alpha_t = \langle \vec{P}_t, \vec{R}_{j-1} \rangle / \|\vec{P}_t\|^2$
        If($\|\alpha_t\| > \|\alpha_{max}\|$)
            Note $\alpha_t$ and update $\alpha_{max}$ : $\alpha_{max} = \alpha_t$;
        Else
            Give a zero value to $\alpha_t$ : $\alpha_t$ =0;
        End if
    End for
Note $\alpha_j : \alpha_j = \alpha_{max}$ ,and update the residual $R_j : R_j = R_{j-1} - \alpha_j P_j$ ;
    Note $P_j(x_i) : P_j(x_i)$ is the prediction label given by the ensemble of classifiers for the ith training sample in jth iteration.
        $P_j(x_i) = \arg\max_{y \in Y} \sum_{t; P_t^*(x_i)=y} \alpha_t, (t = 1, 2, ..., T)$
        Compute the error $e_j$ :
        For i=1 to m
        If $y_i - P_j(x_i) = 0$ then $e_{ji} = 0$, else $e_{ji} = 1$
        End
        $e_j = (\sum_{i=1}^{m} e_{ji}) / m$
        Update the number of iteration: j=j+1;
End while
    Gain the coefficients of the component classifiers $\{\alpha_1, \alpha_2, ..., \alpha_T\}$ and select the classifiers of the nonzero $\alpha_t$ to ensemble.
Output: The combination of the optimal component classifiers:
$P_{opt}(x_i) = \arg\max_{y \in Y} \sum_{t; P_t^*(x_i)=y} \alpha_t, (t = 1, 2, ..., T_{opt}, T_{opt} \le T)$
```

Fig. 8 The MPO-SVME algorithm

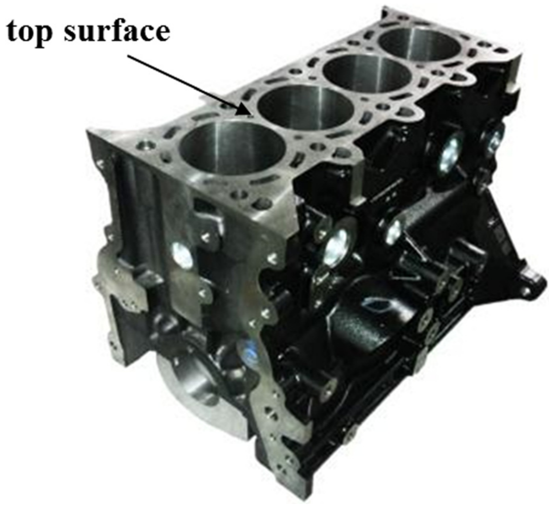


Fig. 9 Engine cylinder blocks processed by a major domestic car manufacturer

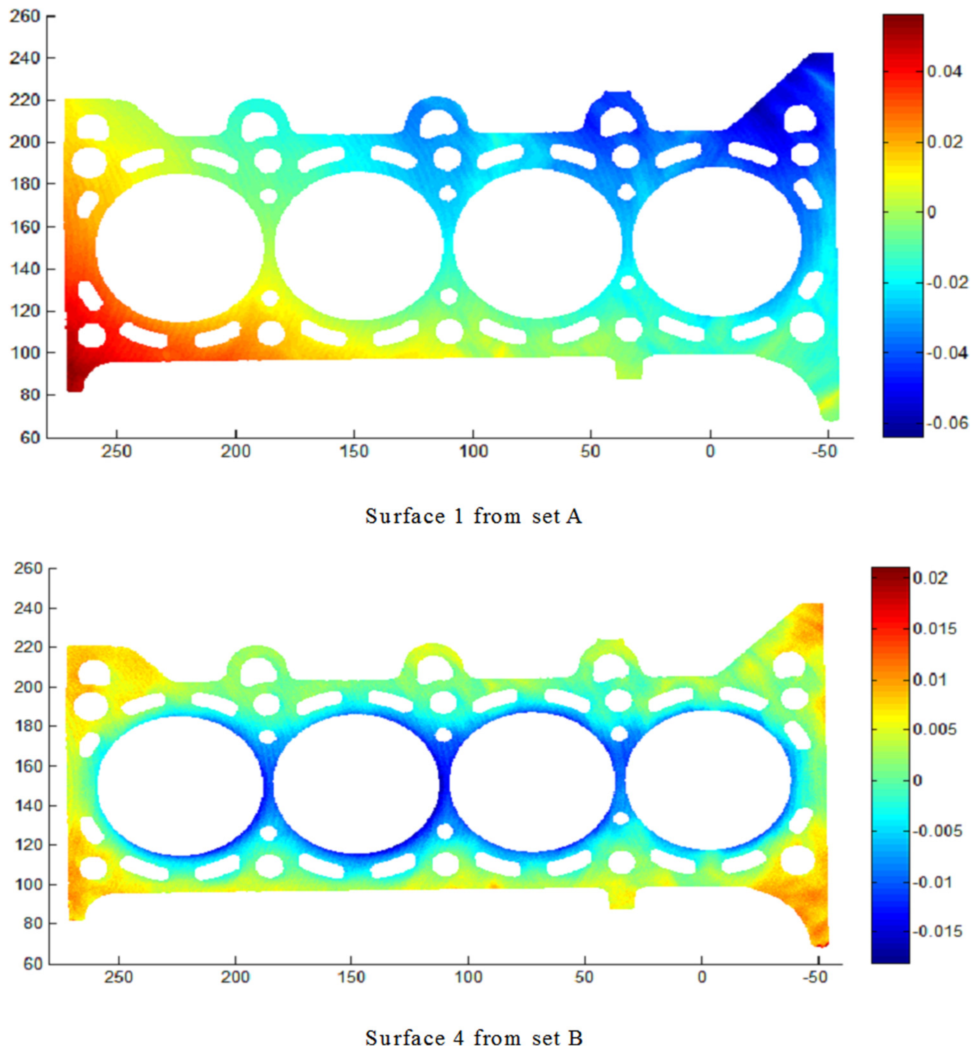
the entire surface is used to extract features because the amount of data representing the entire surface is too large, and it is too time and memory expensive to handle this data for computers. In addition, although only a small area of the surface is used to represent the entire surface, the classification accuracies are sufficiently high in most of the cases when using these 20 square areas for classification purposes, respectively (discussed in Sec. 3.2).

To be specific, six small surfaces from the same position (surface area 16 is chosen as example here) of each surface are chosen with the size of 128 × 128 pixels, which are used as the input of 2D DT-CWT. Figure 12 shows the six surfaces chosen for classification.

It is apparent that the first three surfaces look alike and so do the last three. This is easy to explain, because the first three surfaces are derived from one set while the last three from another set. It is also apparent that surfaces from different sets look different, due to the fact that parts of set A and set B are machined under two different conditions.

Next, each of the surfaces is partitioned into 4 × 4 = 16 nonoverlapping small surfaces with size of 32 × 32 pixels and those stem from the same original surface are considered of the same class (six classes totally). Therefore, 96 samples are obtained for classification and each class has 16 samples. Then, four-level 2D DT-CWT decomposition is implemented on those height data, respectively, feature vectors are generated by computing mean and standard deviation from coefficients of wavelet sub-bands, then each of the 96 samples is represented by a 56-dimensional feature vector($D = 56$)(discussed in Sec. 2.1.2).

3.2 Classification Results. Eight surface samples are randomly selected as training samples and the remaining eight samples as testing samples for every class (As seen in Fig. 13, Nos.



Surface 1 from set A

Surface 4 from set B

Fig. 10 Two samples of the color-coded measurement results from data sets A and B

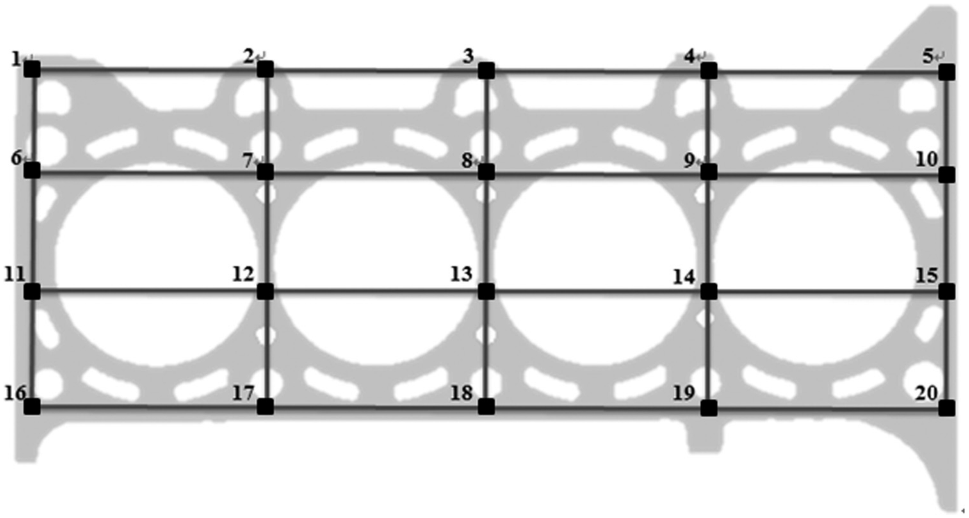
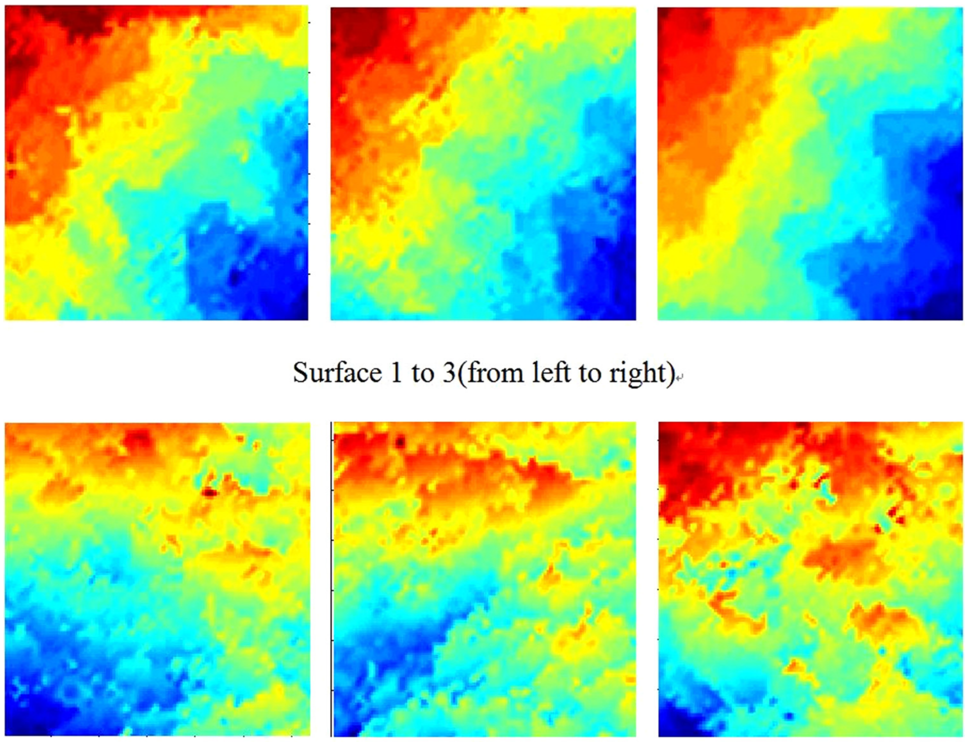


Fig. 11 Selection of small surface samples through a grid chart


Surface 1 to 3(from left to right)

Surface 4 to 6(from left to right)

Fig. 12 Six surfaces selected from set A and set B

1–16 represent 16 surface samples, numbers within the circles represent the randomly selected training samples, the rest represent testing samples), then 48 training samples and 48 testing samples are obtained finally. Gaussian radial basis function (RBF: $K(\mathbf{x}_i, \mathbf{y}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{y}_j\|^2))$ is adopted in this study. The training of classifiers requires the adequate definition of the kernel parameter γ and the penalty parameter C, for the use of inadequate parameter values is likely to result in a less accurate classification. Usually the kernel parameters are determined by a grid-search using n-fold cross-validation. Potential combinations of C and γ are tested in a user-defined range and the best combinations of C and γ are selected based on the results of the cross validation. The optimal value of kernel parameter γ and penalty parameter C is searched by tenfold cross validation experiment, and the parameters with the highest cross validation accuracy are selected (here $\gamma = 8$, $C = 64$).

To study the influence of different decomposition levels on the classification accuracy, levels from 2 to 5 are adopted, respectively. And for comparison purposes, 2D DWT is also implemented to extract features using the same data set. The classification is implemented using the LIBSVM toolbox [48] in a MATLAB environment and MPO-SVME is adopted as the selective ensemble strategy (two-thirds of the features are randomly selected at each level ($d = \frac{2}{3}D$) and the number of component multiclass SVM classifiers is 20 ($T = 20$)), the final classification results are shown in Fig. 14 (CCP (%) means "correct classification percentage," each of the classification results is the average of 30 times).

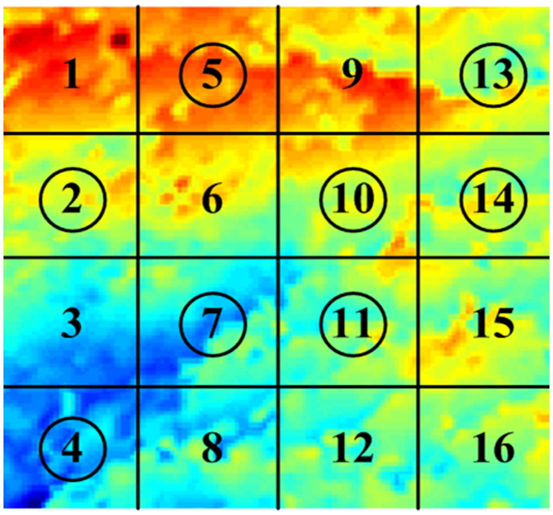


Fig. 13 The generation of training samples and testing samples

As Fig. 14 shows, the classification result of 2D DT-CWT is much better than that of 2D DWT at each level, which demonstrates that the approximate shift invariance and good directional selectivity properties are vital to feature extraction. In regard to the relationship between the value of levels and classification accuracy, it shows that a relatively high CCP is obtained when the number of decomposition levels is four, no matter using DWT or DT-CWT. So four-level 2D DT-CWT decomposition is adopted in this study.

Table 1 presents classification results of single multiclass SVMs, patterns 1–6 represent six different surfaces (1–3 from one set, 4–6 from another). Note that misclassification only occurs within the same set, which means that products machined under the same conditions are quite similar and hard to distinguish.

Table 1 CCPs (%) of single multiclass SVMs

	Classification result						
Pattern	1	2	3	4	5	6	CCP
1	**8**	0	0	0	0	0	100
2	0	**8**	0	0	0	0	100
3	0	**1**	**7**	0	0	0	87.5
4	0	0	0	**5**	**3**	0	62.5
5	0	0	0	0	**7**	**1**	87.5
6	0	0	0	0	0	**8**	100
Average							89.58

The small surfaces from other locations (as shown in Fig. 11, and 20 locations totally) of each of the six surfaces are used to classify, and the results are shown in Fig. 15. Here the upper polygonal line denotes the results of two-class classification identifying each surface sample from set A or from set B, while the lower polygonal line denotes the results of six-class classification that solving the problem of determining which surface each small surface sample originates from. It can be seen that the accuracies of two-class classification are higher than 90% in most of the cases and 100% accuracy is achieved in 11 of the 20 cases, which demonstrates that using a small surface to represent a certain kind of surface is feasible. It is a practical conclusion, since in most of the cases, quality engineers put the greatest emphasis on key positions of the entire workpiece surface, for the quality of these key areas will heavily influence the function of products. For example, the leakage problem is likely to occur in places near the cylinder bore, for the height of these places are relatively low, as the dark blue areas in Surface four (see Fig. 10). So it is acceptable to identify workpiece surfaces based on the classification results of the interested areas. By contrast, the accuracies of six-class classification problems are relatively low, so the effectiveness of using MPO-SVME to increase the classification accuracy is demonstrated in the remaining part of this section.

Figure 16 presents classification results of MPO-SVME for the six engineering surfaces (likewise, surface area 16 is selected from each surface). In order to get reliable results, the experiment are repeated 30 times ($d = 40$ and $T = 20$). The average correct classification rate is 91.74%, which is 2.41% higher than that of single multiclass SVMs. Figure 17 presents the number of selected component classifiers (NSC) at each time ($T = 20$), the average is 2.47 each time. The two figures demonstrate that MPO-SVME can gain a better performance than single multiclass SVMs in classification without using too many classifiers. And this property

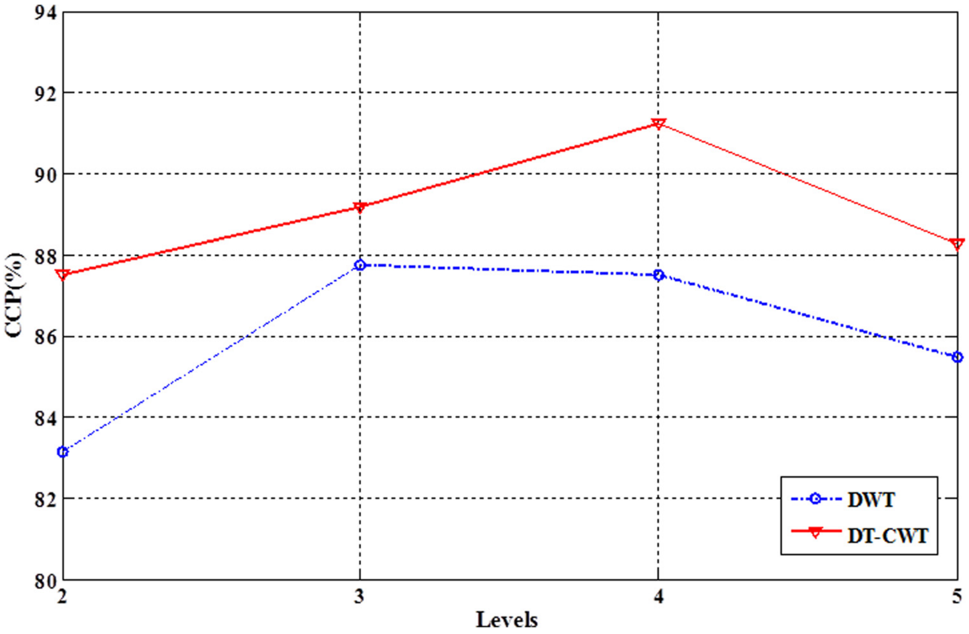


Fig. 14 The CCPs using DT-CWT and DWT with different levels

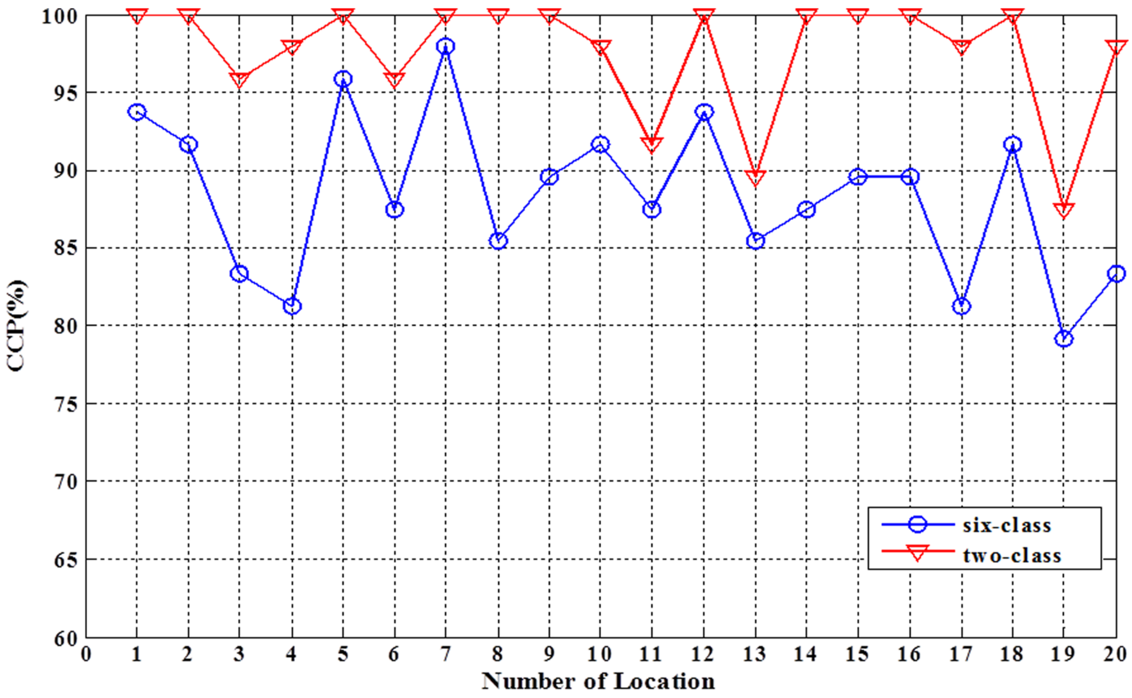


Fig. 15 The CCPs of single multiclass SVMs using surface samples at 20 different locations of each surface

is very useful, because when the quantity of unclassified samples is very large, using too many classifiers will consume a large amount of time, which is not feasible in practical application.

The process of selecting classifiers by MPO-SVME is shown in Fig. 18, and 30 component multiclass classifiers are generated by random subspace algorithm. The upper chart shows the CCP of every component classifier for the original training samples, the middle chart shows the coefficients gained by MPO-SVME, and the lower chart shows the CCP of every component classifier for testing samples. In this selective ensemble, the CCP is 93.75%, which is higher than that of the single multiclass SVM. In the upper and lower charts, the points within black rectangles denote that these component classifiers obtain positive coefficients; the points within red rectangles denote that these component classifiers obtain negative coefficients, and the rest of the points indicate classifiers that gain zero coefficients in the ensemble. The middle chart shows that six component classifiers are selected by MPO-SVME from 30 component classifiers to ensemble (the 2nd, 7th, 9th, 11th, 14th, and 22nd classifiers), and the 14th, 22nd component classifiers obtain greater coefficients than others, due to their better performance in the classification of training samples when compared with others. Consequently, since the classification result of ensemble is equal to the combination of the predictions of all the component classifiers with their coefficients α_t ($t = 1, 2,\ldots,T$), the selected component classifiers are helpful to better the performance of the ensemble due to their higher coefficients.

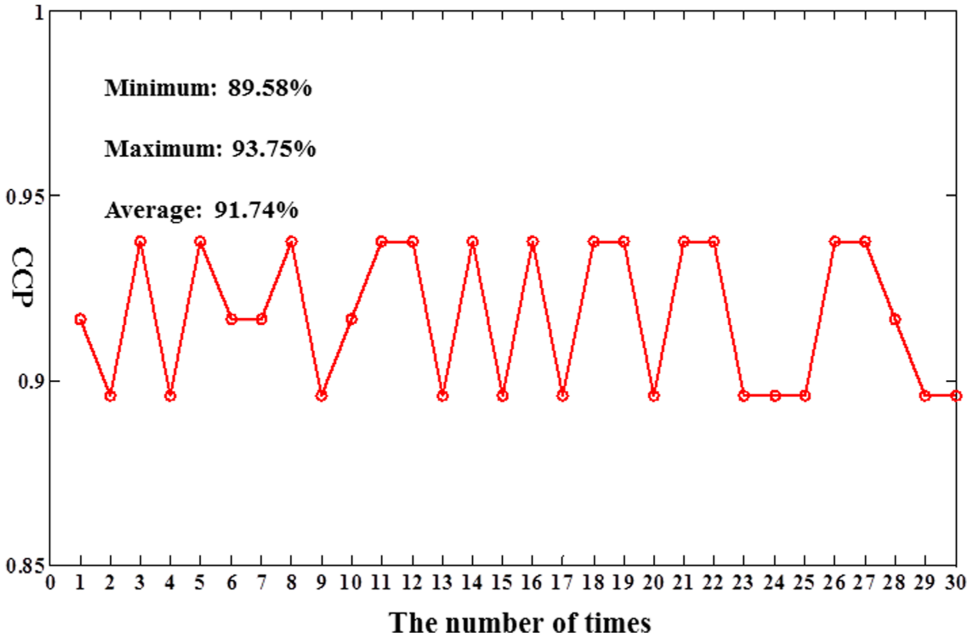


Fig. 16 Classification results of MPO-SVME at each time (30 times totally)

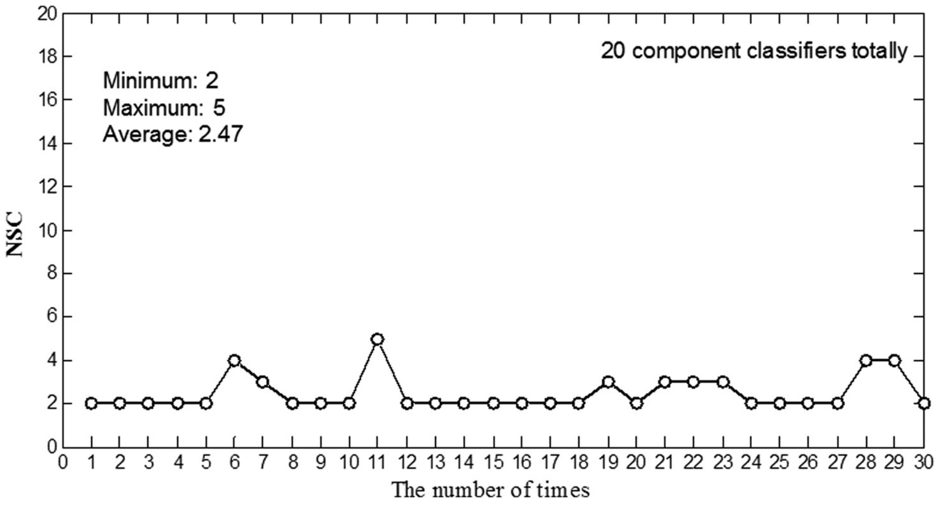


Fig. 17 The NSC at each time

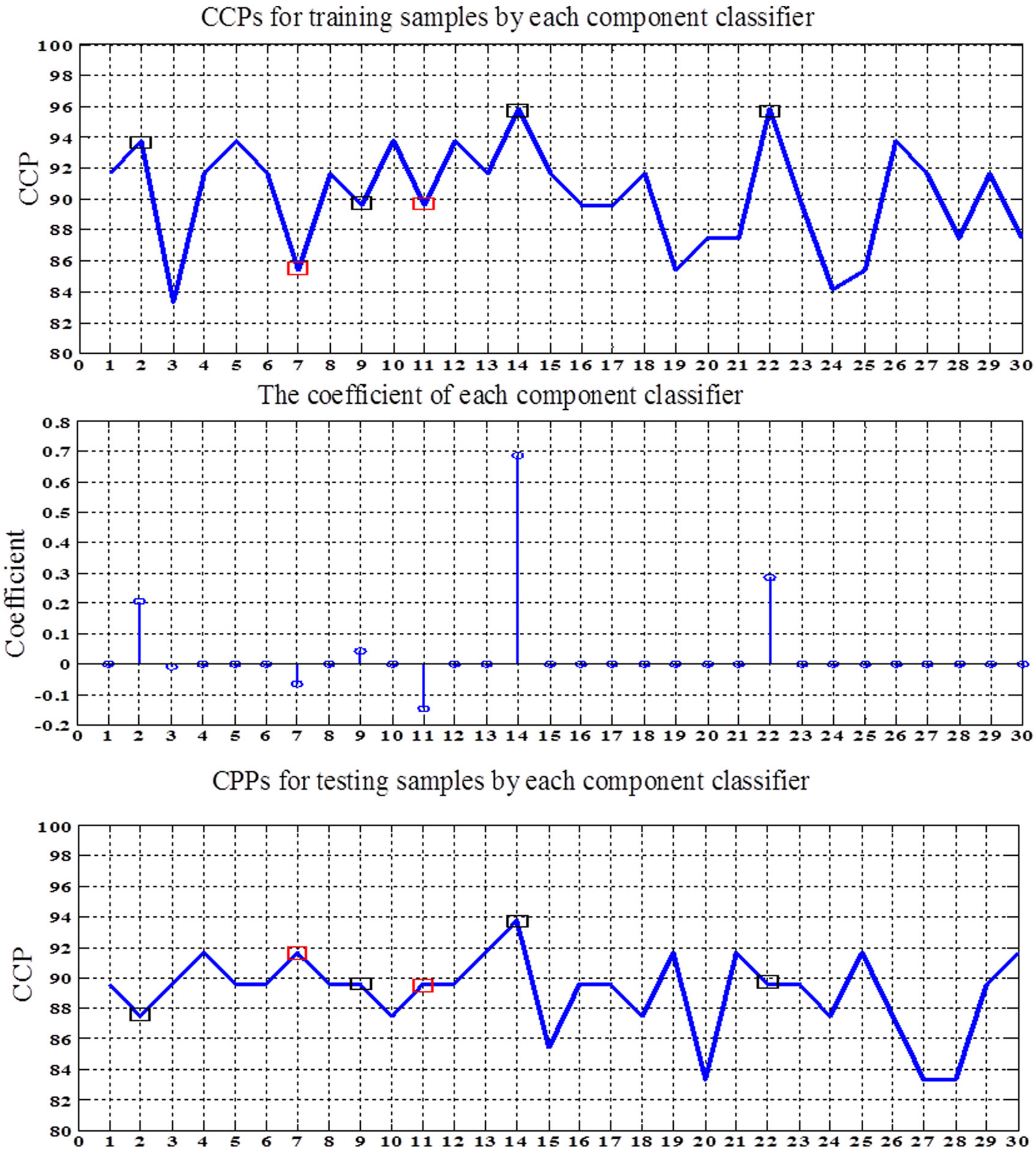


Fig. 18 The process of selective classifiers by MPO-SVME ($T = 30$, $d = 30$)

Table 2 Computational time of the proposed methodology

	Feature extraction[a]	Training[b]	Classification (testing)[c]
Time (unit: second)	0.0111	0.0046	0.0003

[a]Time spent on extracting features of a surface sample using 2D DT-CWT.
[b]Time spent on training classifiers with MPO-SVME (the training set includes 48 surface samples).
[c]Time spent on classifying an unlabeled surface sample (input is feature vector) with the selected classifiers obtained by training.

From the lower chart, it can be seen that there are nine classifiers whose CCPs are lower than 89.58%. If all the component classifiers, including these poor classifiers are treated equally in the classification of testing samples, the performance of the ensemble is likely to be encumbered by these poor classifiers. Hence, MPO-SVME improves the classification performance compared with general ensemble strategies.

Since the proposed surface classification method include several different algorithms in its three steps (feature extraction, training of classifiers, and classification of new surfaces), it is necessary to analyze its time complexity. So the presented method was implemented in a MATLAB R2012a programming environment on a personal computer (PC) with Intel Core2 Duo E6550 central processing unit (CPU) running Windows 7. The training time and testing time of the presented method in case study are given in Table 2, both of which are the average of 30 times. The time spent on feature extraction is also provided in this table. It can be seen that the presented method cost very little time in feature extraction and training, and the time spent on classifying a surface sample can even be ignored. Therefore, from viewpoint of computational cost, the presented method is efficient to be applied to a practical manufacturing system.

3.3 Sensitivity Analysis

3.3.1 Sensitivity Analysis to the Number of Selected Features in MPO-SVME. In MPO-SVME, one important parameter to be determined is how many features should be selected in the random subspace algorithm, due to the fact that the hyperplanes are the

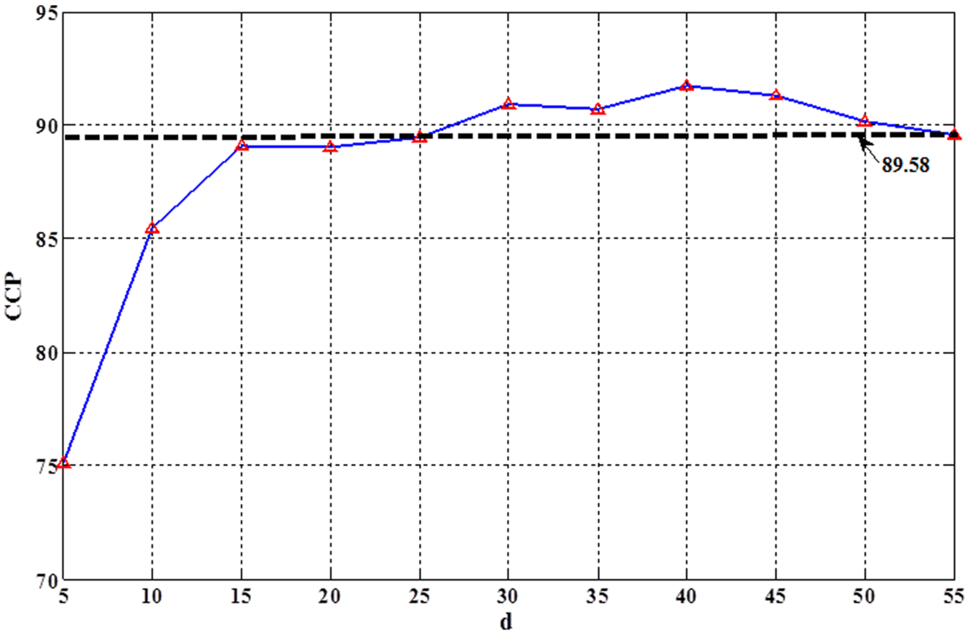


Fig. 19 Comparison of CCPs with different numbers of random features (*T* = 20)

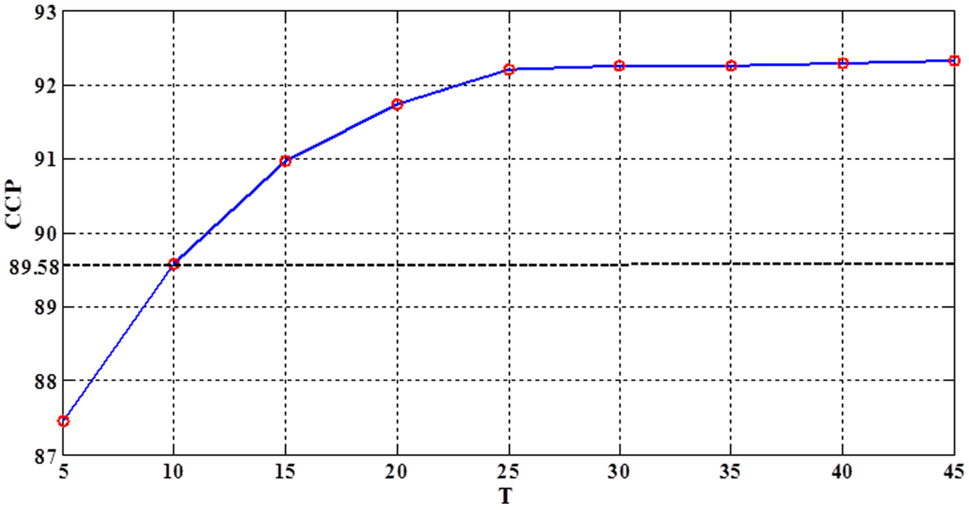


Fig. 20 Comparison of CCPs for different multiclass SVM classifier *T* (*n* = 40)

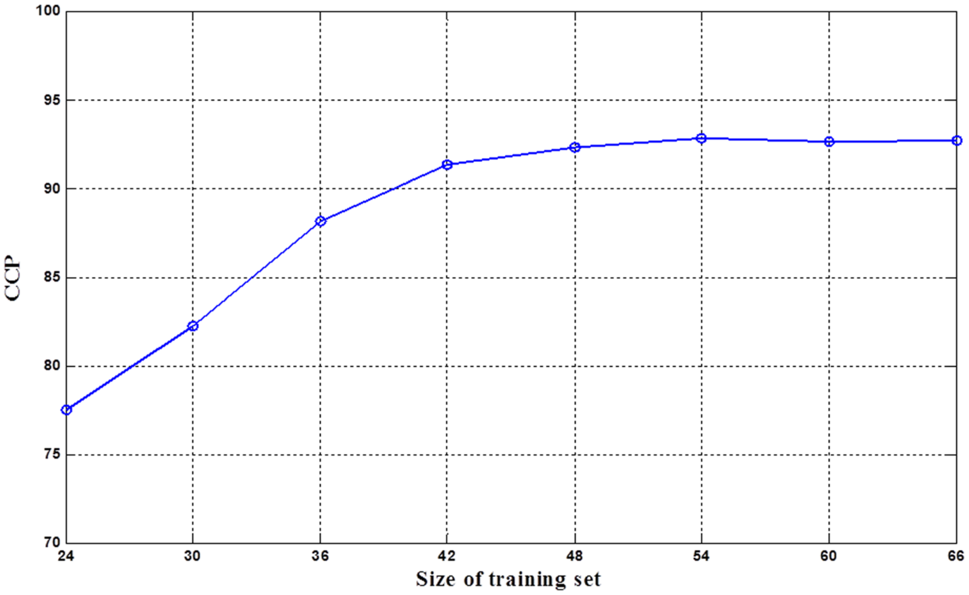


Fig. 21 CCPs for different sizes of training set

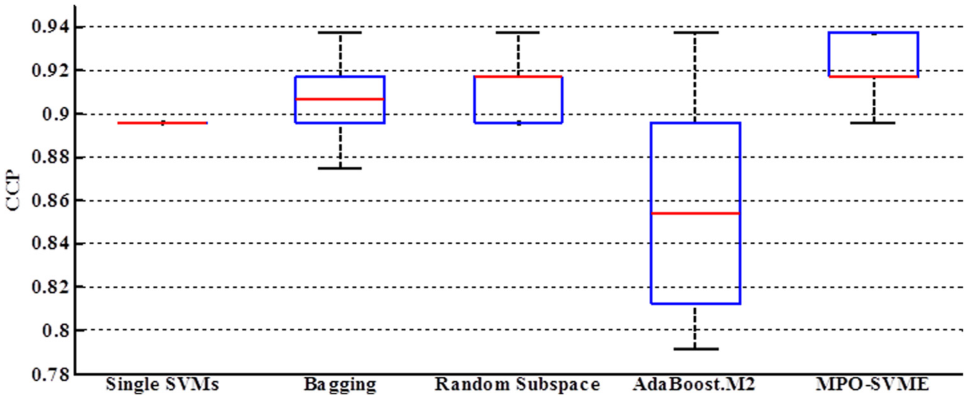


Fig. 22 Boxplots of the classification results using different strategies

functions of the selected features, so the number of randomly selected features could influence the classification results. Figure 19 shows the classification results of MPO-SVME with different numbers of selected features (56 features totally, and it should be noted that all the classification results in Sec. 3.3 are the average of 30 times).It can be found that increasing the number of features can substantially improve the performance of MPO-SVME when d is relatively small (less than 15). As d continues to increase, the CCP will grow slowly and reach the maximum when d is 40. Hence, selecting about 40 features from the total 56 features is appropriate to get the optimal result.

3.3.2 Sensitivity Analysis to the Number of Component Classifiers. T The number of component multiclass SVM classifier T influences the performance of SVM ensemble. Figure 20 presents CCPs for different numbers of component multiclass SVM classifier T. It shows that increasing the number of T can improve the performance of MPO-SVME up to a certain level of accuracy. This may be explained by the fact that it is more likely to generate some preferable component classifiers that are available for selection when T is larger. But this is not to say bigger is better, since any further increase of the number of T after reaching such a limit will not obviously improve the performance of MPO-SVME, and a larger T will cost more time in training SVMs and decreases the effectiveness of classifying engineering surfaces as quickly as possible. Therefore, 25 component multiclass SVM classifiers are enough to get the optimal results.

3.3.3 Sensitivity Analysis to the Size of Training Set. As 96 surface samples are generated in case study and 16 samples for each class, 30 surface samples are randomly selected as testing samples (five training samples from each class), then different numbers of training samples are randomly selected from the remaining ones and the number of the selected training samples is the same among each class. The experiment is repeated 30 times ($d = 40$ and $T = 20$). From Fig. 21, it can be found that increasing the training samples can improve the performance of MPO-SVME up to a certain level of accuracy. This could be explained by the fact that there is a better chance of true representation of a problem space with enough training sets. However, any further increase of the training size after reaching such limits will not significantly improve the performance of MPO-SVME. Moreover, the larger training set results in higher time on training classifiers.

3.3.4 Comparison Analysis for MPO-SVME, Single SVMs, and Ensemble All. In order to further evaluate the performance of MPO-SVME, it is compared with other SVMs strategies including single SVMs and Ensemble All (i.e., ensemble all the component multiclass classifiers $C_1, C_2, C_3, ..., C_T$). And the classification results of each strategy are shown in Fig. 22 and Table 3 (in Ensemble All strategy (including bagging [39], random subspace

Table 3 CCPs of different strategies (average of 30 times)

		Ensemble all			Selective ensemble
Strategy	Single SVMs	Bagging	Random subspace	AdaBoost.M2	MPO-SVME
CCP(%)	89.58	90.69	90.97	85.56	92.22

[41], and Adaboost.M2 [49]), $T=25$; in MPO-SVME, $T=25$, and $d=40$). The boxplots in Fig. 22 reflect the distributions of the classification results of each approach, for all the approaches are repeated 30 times. It can be seen that random subspace outperforms other two Ensemble All strategies, which denotes that random subspace is more applicable to the classification of engineering surfaces in this study. It is worth noting that MPO-SVME outperforms Single SVMs and Ensemble All strategies in most of the 30 cases, and according to Table 3, MPO-SVME achieves the highest average CCPs in all the methods. This proves that MPO-SVME has better performance when comparing with those commonly used methods such as single SVMs and Ensemble All.

4 Conclusion

A method consists of feature extraction and classification is proposed to classify engineering surfaces using high definition metrology. In this two-step method, DT-CWT is used to extract features and MPO-SVME is used for the purpose of classification. A case study is conducted in this paper, and it first demonstrates the effectiveness of using DT-CWT for feature extraction. The results indicate that DT-CWT outperforms DWT in feature extraction, and this can be explained by the fact that DT-CWT has approximate shift invariance and good directional selectivity properties whereas DWT does not. Then the performance of MPO-SVME in classification is validated using real-world data. The results demonstrate that MPO-SVME shows improved generalization performance by assembling some accurate and diverse component multiclass SVMs. Furthermore, the influences of some key parameters of MPO-SVME upon its performance are analyzed, which aims to find the suitable parameters for constructing MPO-SVME. In general, the proposed method based on DT-CWT and MPO-SVME is effective and efficient in classifying engineering surfaces, especially surfaces with clear surface patterns.

Acknowledgment

The authors greatly acknowledge the editor and the referees for their valuable comments and suggestions that have led to a substantial improvement of the paper. This work was supported by the National Natural Science Foundation of China (Grant No. 51275558 and 50905114), National Key Science and Technology Research Program of China (Grant No. 2014ZX04015-021), and Shanghai Rising-Star Program (Grant No. 13QA1402100).

References

[1] Whitehouse, D., 1997, "Surface Metrology," Meas. Sci. Technol., **8**(9), pp. 955–972.

[2] Whitehouse, D., 1982, "The Parameter Rash—Is There a Cure?" Wear, **83**(1), pp. 75–78.

[3] International Organization for Standardization ISO 4287, 1997, *Geometrical Product Specifications (GPS)—Surface Texture: Profile Method—Terms, Definitions and Surface Texture Parameters*, ISO, Geneva.

[4] ASME B46.1, 2002, *Surface Texture (Surface Roughness, Waviness, and Lay)* ASME, New York.

[5] Leopold, J., Günther, H., and Leopold, R., 2003, "New Developments in Fast 3D-Surface Quality Control," Measurement, **33**(2), pp. 179–187.

[6] Nguyen, H. T., Wang, H., and Hu, S. J., 2013, "Characterization of Cutting Force Induced Surface Shape Variation Using High-Definition Metrology," ASME, J. Manuf. Sci. Eng., **135**(4), p. 041014.

[7] Blunt, L., and Jiang, X., 2003, *Advanced Techniques for Assessment Surface Topography: Development of a Basis for 3D Surface Texture Standards*, Kogan Page Science, London.

[8] Graziano, A. A., Ganguly, V., Schmitz, T., and Yamaguchi, H., 2014, "Control of Lay on Cobalt Chromium Alloy Finished Surfaces Using Magnetic Abrasive Finishing and Its Effect on Wettability," ASME, J. Manuf. Sci. Eng., **136**(3), p. 031016.

[9] Wit, G., and Krzysztof, Ż., 2014, "Characterization of Surface Integrity Produced by Sequential Dry Hard Turning and Ball Burnishing Operations," ASME, J. Manuf. Sci. Eng., **136**(3), p. 031017.

[10] VenkatRamana, K., and Ramamoorthy, B., 1996, "Statistical Methods to Compare the Texture Features of Machined Surfaces," Pattern Recognit., **29**(9), pp. 1447–1459.

[11] Dong, W., Sullivan, P., and Stout, K., 1994, "Comprehensive Study of Parameters for Characterizing Three-Dimensional Surface Topography: IV: Parameters for Characterizing Spatial and Hybrid Properties," Wear, **178**(1), pp. 45–60.

[12] Satoh, G., Huang, X., Ramirez, A. G., and Yao, Y. L., 2012, "Characterization and Prediction of Texture in Laser Annealed NiTi Shape Memory Thin Films," ASME, J. Manuf. Sci. Eng., **134**(5), p. 051006.

[13] Tsa, D.-M., and Wu, S.-K., 2000, "Automated Surface Inspection Using Gabor Filters," Int. J. Adv. Manuf. Technol., **16**(7), pp. 474–482.

[14] Zhang, M., Levina, E., Djurdjanovic, D., and Ni, J., 2008, "Estimating Distribution of Surface Parameters for Classification Purposes," ASME, J. Manuf. Sci. Eng., **130**(4), p. 031010.

[15] Fu, S., Muralikrishnan, B., and Raja, J., 2003, "Engineering Surface Analysis with Different Wavelet Bases," ASME, J. Manuf. Sci. Eng., **125**(4), pp. 844–852.

[16] Liao, Y., Stephenson, D. A., and Ni, J., 2012, "Multiple-Scale Wavelet Decomposition, 3D Surface Feature Exaction and Applications," ASME, J. Manuf. Sci. Eng., **134**(1), p. 11005.

[17] Li, Y., and Ni, J., 2011, "B-Spline Wavelet-Based Multiresolution Analysis of Surface Texture in End-Milling of Aluminum," ASME, J. Manuf. Sci. Eng., **133**(1), p. 011014.

[18] Yu, J., 2012, "Machine Tool Condition Monitoring Based on an Adaptive Gaussian Mixture Model," ASME, J. Manuf. Sci. Eng., **134**(3), p. 031004.

[19] Raja, J., Muralikrishnan, B., and Fu, S., 2002, "Recent Advances in Separation of Roughness, Waviness and Form," Precis. Eng., **26**(2), pp. 222–235.

[20] Badashah, S. J., and Subbaiah, P., 2011, "Image Enhancement and Surface Roughness With Feature Extraction Using DWT," International Conference on Sustainable Energy and Intelligent Systems, IET, Chennai, India, July 20–22, pp. 754–759.

[21] Kingsbury, N., 1999, "Image Processing With Complex Wavelets," Philos. Trans. R. Soc. London, **357**(1760), pp. 2543–2560.

[22] Kingsbury, N., 1998, "The Dual-Tree Complex Wavelet Transform: A New Technique for Shift Invariance and Directional Filters," Proceedings of the 8th IEEE Digital Signal Processing Workshop, IEEE, Bryce Canyon, UT, Aug. 9–12, pp. 120–131.

[23] Kingsbury, N., 2001, "Complex Wavelets for Shift Invariant Analysis and Filtering of Signals," Appl. Comput. Harmonic Anal., **10**(3), pp. 234–253.

[24] Selesnick, I., Baraniuk, R., and Kingsbury, N., 2005, "The Dual-Tree Complex Wavelet Transform," IEEE Signal Process. Mag., **22**(6), pp. 123–151.

[25] Vapnik, V., 1999, *The Nature of Statistical Learning Theory*, Springer, New York.

[26] Gunn, S. R., 1998, "Support Vector Machines for Classification and Regression," ISIS Technical Report No. 14.

[27] Cristianini, N., and Shawe-Taylor, J., 2000, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University, Cambridge, UK.

[28] Paredis, C. J., 2010, "Using Support Vector Machines to Formalize the Valid Input Domain of Predictive Models in Systems Design Problems," ASME J. Mech. Des., **132**(10), p. 101001.

[29] Du, S., Lv, J., and Xi, L., 2012, "On-Line Classifying Process Mean Shifts in Multivariate Control Charts Based on Multiclass Support Vector Machines," Int. J. Prod. Res., **50**(22), pp. 6288–6310.

[30] Pal, S., and Pal, A., 2001, *Pattern Recognition: From Classical to Modern Approaches*, World Scientific Publishing Company, Singapore, pp. 427–451.

[31] Ho, T., 2000, "Complexity of Classification Problems and Comparative Advantages of Combined Classifiers," Multi. Classifier Syst., **1857**, pp. 97–106.

[32] Kim, H. C., Pang, S., Je, H.-M., Kim, D., and Yang Bang, S., 2003, "Constructing Support Vector Machine Ensemble," Pattern Recognit., **36**(12), pp. 2757–2767.

[33] Wang, S. J., Mathew, A., Chen, Y., Xi, L. F., Ma, L., and Lee, J., 2009, "Empirical Analysis of Support Vector Machine Ensemble Classifiers," Expert Syst. Appl., **36**(3), pp. 6466–6476.

[34] Pang, S., Kim, D., and Bang, S. Y., 2005, "Face Membership Authentication Using SVM Classification Tree Generated by Membership-Based LLE Data Partition," IEEE Trans. Neural Networks, **16**(2), pp. 436–446.

[35] Lei, Z., Yang, Y., and Wu, Z., 2006, "Ensemble of Support Vector Machine for Text-Independent Speaker Recognition," Int. J. Comput. Sci. Networks Secur., **6**(5), pp. 163–167.

[36] Hung, C., and Chen, J. H., 2009, "A Selective Ensemble Based on Expected Probabilities for Bankruptcy Prediction," Expert Syst. Appl., **36**(3), pp. 5297–5303.

[37] Jaya Priya, K., and Rajesh, R., 2010, "Local Fusion of Complex Dual-Tree Wavelet Coefficients Based Face Recognition for Single Sample Problem," Procedia Comput. Sci., **2**, pp. 94–100.

[38] Hsu, C. W., and Lin, C. J., 2002, "A Comparison of Methods for Multiclass Support Vector Machines," IEEE Trans. Neural Networks, **13**(2), pp. 415–425.

[39] Breiman, L., 1996, "Bagging Predictors," Mach. Learn., **24**(2), pp. 123–140.

[40] Schapire, R. E., 1990, "The Strength of Weak Learnability," Mach. Learn., **5**(2), pp. 197–227.

[41] Ho, T. K., 1998, "The Random Subspace Method for Constructing Decision Forests," IEEE Trans. Pattern Anal. Mach. Intell., **20**(8), pp. 832–844.

[42] Zhou, Z.-H., Wu, J., and Tang, W., 2002, "Ensembling Neural Networks: Many Could be Better Than All," Artif. Intell., **137**(1), pp. 239–263.

[43] Partalas, I., Tsoumakas, G., Katakis, I., and Vlahavas, I., 2006, "Ensemble Pruning Using Reinforcement Learning,"Adv. Artif. Intell., **3955**(1), pp. 301–310.

[44] Giacinto, G., Roli, F., and Fumera, G., 2000, "Design of Effective Multiple Classifier Systems by Clustering of Classifiers," Proceedings of the 15th International Conference on Pattern Recognition, Barcelona, September 3–7, Vol. 2, pp. 160–163.

[45] Martınez-Munoz, G., and Suárez, A., 2006, "Pruning in Ordered Bagging Ensembles," Proceedings of the 23rd International Conference on Machine Learning, ACM, New York, June 25–29, pp. 609–616.

[46] Mao, S., Jiao, L., Xiong, L., and Gou, S., 2011, "Greedy Optimization Classifiers Ensemble Based on Diversity," Pattern Recognit., **44**(6), pp. 1245–1261.

[47] Coherix, http://www.coherix.com

[48] Chang, C. C., and Lin, C. J., 2011, "LIBSVM: A Library for Support Vector Machines," ACM Trans. Intell. Syst. Technol., **2**(3), p. 27.

[49] Freund, Y., and Schapire, R., 1995, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," Proceedings of the Second European Conference on Computer and System Sciences, Springer, Berlin-Heidelberg, Germany, March 13–15, pp. 23–37.