ON-LINE CLASSIFICATION OF SURFACE DEFECTS IN HOT ROLLING PROCESSES

Qingyu Yang, Qiang Li and Judy Jin
Department of Industrial and Operations Engineering
University of Michigan
Ann Arbor, Michigan

Tzyy-Shuh Chang
OG Technologies, Inc.
Ann Arbor, Michigan

KEYWORDS

Rolling process, Feature Selection, Genetic Algorithm, Support Vector Machine (SVM)

ABSTRACT

This paper develops a methodology for the on-line classification of different types of surface defects in rolling processes. The method developed consists of three major steps: feature generation, feature selection, and classifier design. Based on the engineering knowledge of rolling processes, a set of potential features are generated from the local regions of sensing images that contain the suspicious surface defects. Then, a feature selection method based on the Genetic Algorithm (GA) and a linear classifier is developed to eliminate redundant features. Finally, a new classifier, named weighted hierarchical SVM, is proposed to classify different types of surface defects. This newly designed classifier can improve the classification accuracy for those defects that have similar shapes or/and have small sample sizes. A case study is conducted in a real world rolling process to demonstrate the implementation procedures and the effectiveness of the developed methods.

INTRODUCTION

Rolling process is a high-speed bulk deformation process that reduces the thickness of, or changes the cross-section of a long workpiece by applying compressive forces through a set of rolls. Surface defects are critical quality concerns in the rolling industry (Jin et al. 2008). Four major types of surface defects exist in rolling processes: blob, longitudinal, repeating, and transverse. Each of these types of surface defects is further divided into the corresponding subtypes. For example, the longitudinal type of defect can be further categorized as burnt steel, cross roll, lap, overfill, scratch, or seam. In general, each subtype of surface defects corresponds to a manufacturing problem, such as material overfills, nonmetallic inclusion, or porosities. As a result, the accurate detection and classification of the surface defects can help discover the major causes of the surface defects for expediting the process correction decisions.

For detecting surface defects in hot rolling processes (Okamoto et al. 1988; Sugimoto and Kawaguchi 1998), traditional methods used infrared detectors to monitor the temperature difference between the defective regions and the non-defective regions. However, those traditional detection methods did not perform satisfactorily in practice, because the temperature difference was not large enough under harsh environmental conditions due to dust, water sprays, and lubricants, etc. In recent years, with the development of advanced imaging technologies, vision sensors have been feasibly adopted at the hot rolling process to online collect product surface images. Therefore, on-line detection and classification of surface defects becomes possible.

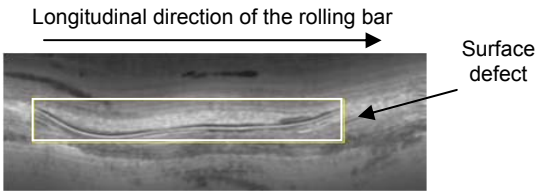

FIGURE 1. REAL TIME SENSING IMAGE OF A HOT BAR IN A ROLLING PROCESS

The challenges of automatic detection and classification of surface defects based on sensing images are twofold: low signal-to-noise ratio and random noisy background. Figure 1 shows an example of a portion of a sensing image from a hot bar in a rolling process. It can be seen that the surface defects usually mingle with surface marks and material impurities. Very little research focusing on this topic has been published. Li et al. (Li et al. 2007) developed a method to detect the "seam" surface defects efficiently using snake projection and discrete wavelets. In their research, only one type of surface defects, seam, is studied; the classification of multiple types of surface defects is not fully considered.

A two-stage methodology for the detection/classification of rolling surface defects is developed in this research. At the first stage, the Soble edge detector (Ziou and Tabbone 1998) used by Li's group (Li et al. 2007), and the MSER region detector (Sonka et al. 1999) are adapted to rapidly screen out the suspicious local regions that may contain the defects from all the sensing images. Because the Soble edge detector and the MSER region detector are well documented in the literature, this paper focuses only on the second stage, i.e., the classification of the surface defects based on the suspicious local regions that are obtained from the first screening stage. The remainder of this paper is organized as follows. In the next section, the details of the methodology development will be presented, in which three major steps of feature generation, feature selection, and classifier design will be discussed. The proposed methods are then implemented and illustrated through a case study in a real world rolling process. Finally, a conclusion is provided.

CLASSIFICATION METHOD

There are three major steps for the methodology development, which are feature generation, feature selection, and classifier design. As shown in Figure 2, feature generation is conducted at the first step, in which a set of potential features is generated from the suspicious defect regions of sensing images based on the engineering knowledge. Then, a feature selection method based on GA and a linear classifier is developed to eliminate redundant features. Finally, based on the selected features, classifiers are designed for defect classification. The weighted classifier is used to increase the classification accuracy for surface defects that have small sample sizes. A new classifier, named weighted hierarchical classifier, is also proposed to further improve the classification accuracy for surface defects that have similar shapes.

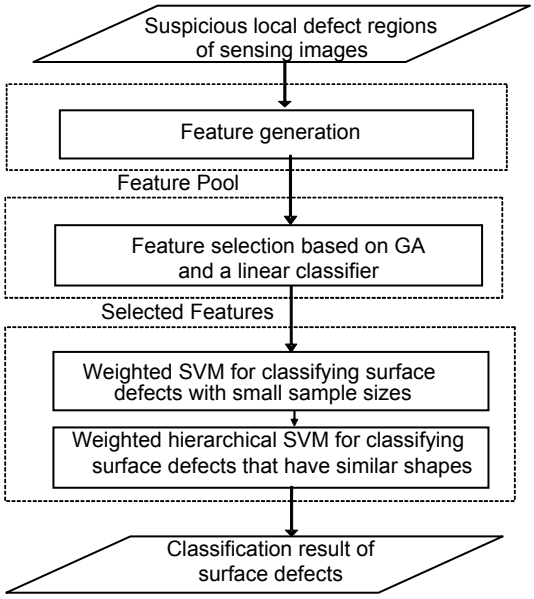

FIGURE 2. METHODOLOGY OF CLASSIFICAITON METHOD BASED ON SUSPICIOUS DEFECT REGIONS OF SENSING IMAGES.

Feature Generation

Feature generation is a method to generate a set of low dimensional features that can characterize high dimensional data. A set of n features, denoted as $\mathbf{F}_0 = [f_1, f_2, ..., f_n]$, is used in this paper to represent the potential defect regions of sensing image data. In general, sensing images taken for the rolling process may have a low signal to noise ratio, and the number of the surface defects could be large. Thus, it is difficult to intuitively determine the exact features that can effectively distinguish the surface defects in the rolling processes. In this

research, 50 potential features are initially generated as a feature pool based on the engineering knowledge of image properties, statistical quantity, and shape description of the local defect region of the sensing images in rolling processes.

The features generated based on the engineering knowledge may include redundant information, which not only increases the computational burden, but also hinders the classification performance. Therefore, a feature selection method based on GA and a linear classifier is developed in the next subsection, which is used to select the most efficient features from the initial feature pool.

Feature Selection Based on GA and a Linear Classifier

Feature selection is a process used to choose a subset of the optimal features $\mathbf{F}_s$ from the feature pool $\mathbf{F}_0, \mathbf{F}_s \subseteq \mathbf{F}_0$, which can preserve the most useful information for the classification. In this research, the GA is applied for the feature selection purpose.

The GA (Holland 1992) simulates the natural evolutionary process to find either the exact or approximate solutions to search problems. GAs are generally implemented as a computer simulation in which a population of abstract representations (called chromosomes) of candidate solutions (called individuals) evolves toward better solutions. The evolution usually starts from a population of randomly generated individuals. In each generation, multiple individuals are randomly selected as "parents" that will reproduce new individuals by using genetic operators (crossover and mutation). The fitness of each individual is calculated and the individuals with large fitness values generate a new generation, which is then used in the next iteration of the algorithm. The GA commonly terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

The implementation of GA for the feature selection is illustrated in Figure 3. The first step of this algorithm is to randomly generate a set of t chromosomes that are the abstract representation of potential solutions of selected features. A chromosome, denoted as $\mathbf{g}_k = [g_{k,1},...,g_{k,i},...,g_{k,n}]$,where k=1,2,…,t, i=1,2,…,n; is encoded as follows:

$$g_{k,i} = \begin{cases} 1 & \text{if the } i^{th} \text{ feature is selected} \\ 0 & \text{o.w.} \end{cases} \quad (1)$$

Then, at the second step, a pair of chromosomes is randomly selected from the previous generation as parents to reproduce a pair of children through genetic operators: crossover and mutation.

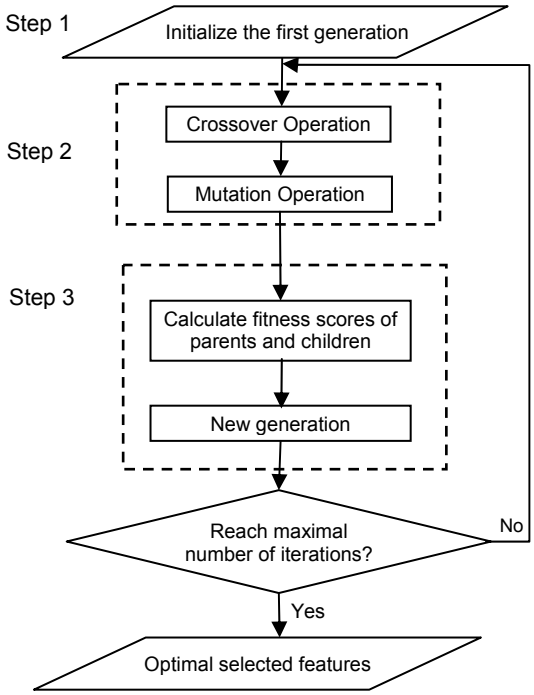

FIGURE 3. ALGORITHM 1 FOR FEATURE SELECTION BASED ON GA AND A LINEAR CLASSIFIER.

Crossover: a crossover operator reproduces a pair of children, $\mathbf{g}_1$ and $\mathbf{g}_2$, from a set of parents, $\mathbf{g}_3$ and $\mathbf{g}_4$. It involves a probability that the genes of the two parents' chromosomes, $g_{3,i}$ and $g_{4,i}$, i=1,2,..,n, are swapped. In the developed algorithm, $\mathbf{g}_1$ and $\mathbf{g}_2$ are reproduced based on the generation of n random variables that are [0,1] uniformly distributed, denoted as r_i, i=1,2,..,n, using the following rules:

$$g_{1,i} = \begin{cases} g_{3,i} & \text{if } r_i > 0.5 \\ g_{4,i} & \text{o.w.} \end{cases} \quad (2)$$

$$g_{2,i} = \begin{cases} g_{4,i} & \text{if } r_i > 0.5 \\ g_{3,i} & \text{o.w.} \end{cases} \quad (3)$$

where i=1,2,…,n.

Mutation: a genetic operator is used to maintain genetic diversity from one generation of chromosomes to the next. It involves a probability that an arbitrary gene, $g_{k,i}$, k=1,2,…,t, i=1,2,…,n, in the chromosome of the child will be changed from its original state. In the developed algorithm, the new [0,1] uniformly distributed random variables, q_i, i=1,2,..,n, are generated. Based on this outcome, the mutation operation is implemented using the following rules:

$$g_{k,i} = \begin{cases} 1 - g_{k,i} & \text{if } q_i > 0.9 \\ g_{k,i} & \text{o.w.} \end{cases} \quad (4)$$

where i=1,2,…,n; k=1,2,…,t.

Finally, at the third step, a new generation of the population is selected from both the parents and children based on their fitness evaluation. The fitness function of a chromosome is based on the classification error rate, when the selected features represented by the chromosome are used to classify the defects of the training data. Because GA is an iterative method and the fitness function needs to be evaluated for each chromosome for both parents and children at each iterative step, the computation time is tremendous and unrealistic if the fitness function is based on a complex classifier. As a result, the fitness function implemented in the developed algorithm is based on a simpler classifier, the linear classifier (Duda et al. 2001). For the detailed description of a linear classifier, please refer to Duda's work (Duda et al. 2001). Suppose the classification accuracy of the linear classifier is ε. The fitness score, denoted as f, is calculated based on the following fitness function:

$$f = \frac{1}{(1-\varepsilon)^3} \quad (5)$$

In the new generation of the population, h individuals are selected from the parents that have the highest fitness score; the remaining t-h individuals are selected from the children that also have the highest fitness score. In the application of the GA method t is chosen as 30 and h is selected as 2. The generational process is repeated until the termination condition has been reached, that is, when a maximum number of 30,000 generations have been produced.

Classifier Design

The design of a good classifier is to accurately classify the surface defects using the features selected in the previous subsection. Although the linear classifier is used in the feature selection process to calculate the fitness value to reduce the computational constrains, a Support Vector Machine (SVM) (Vapnik 1998) is used in the final classification step to enhance the nonlinear classification capability. This significantly increases the final classification performance. The SVM is formulated as follows:

$$\begin{aligned} &\text{Min} \sum_{w,b,\xi} \frac{1}{2}\|L\|^2 + C\sum_{i=1}^{N}\xi_i \quad (6) \\ &s.t. \quad y_i(L\cdot\phi(x_i)+b) \geq 1-\xi_i \\ &\xi_i \geq 0, i = 1, 2,...N \end{aligned}$$

where $\phi(x_i)$ is a nonlinear mapping of the input data x_i into the high dimensional feature space; $L\cdot\phi(x_i)+b=0$ determines the optimal separating hyper-planes; ξ_i, i=1,2,…,N, are the slack variables; N is the total sample size of all classes, and C is a penalty parameter to be chosen by the user, with a larger C corresponding to assigning a higher penalty to misclassification. For a detailed description of the SVM method, please refer to (Vapnik 1998).

With the application of the SVM classifier, the overall classification accuracy for classifying the surface defects in the rolling processes can be satisfied. However, the misclassification error rate could be large for some specific classes. This could be for two reasons: (1) the number of occurrences for some surface defects could be very small in a real world rolling process. Therefore, the sample sizes of these defects are small, resulting in large classification error rate; and (2) some classes have similar shapes, thus the extracted features may be overlapping among these classes.

In order to tackle these two problems, a new classifier, named weighed hierarchical SVM, is developed in this paper. It extends the weighted SVM to a classifier with a hierarchical structure. An algorithm that can automatically find the hierarchical structure is further developed.

Weighted SVM. For the SVM method, it is found in the literature that when the sample size of a particular class is extremely small, the misclassification error rate for this class will increase. The weighted SVM classifier is developed (Huang and Du 2005) to solve this problem by assigning a different penalty for each

of the classes based on their sample sizes. The formulation of the weighted SVM is listed as follows:

$$\text{Min} \sum_{w,b,\xi} \frac{1}{2}\|L\|^2 + C\sum_{i=1}^{N} z_i \xi_i \qquad (7)$$

$$s.t. \quad y_i(L \cdot \phi(x_i) + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0, i = 1, 2, ...N$$

where z_i is a weighting factor for the i^{th} training sample; the other notations are the same as those in Equation 6.

In this paper, the weighting factor, z_i, is calculated as Equation (8).

$$z_i = \begin{cases} \dfrac{N}{m_i n} & \text{if } \dfrac{m_i}{N} \leq 0.01 \\ 0 & \text{o.w.} \end{cases} \qquad (8)$$

where N is the total sample size of all classes; n is the number of classes; and m_i, i=1,2,…,n. indicates the sample size of class i.

Weighted Hierarchical SVM. The weighted hierarchical SVM classifier is proposed to improve the classification accuracy of surface defects that have similar shapes and thus are difficult to be distinguished. The main idea of the hierarchical classifier is to classify the defects at two levels. At the higher level, classes of surface defects that are difficult to be distinguished are grouped as mixed classes. (The algorithm to generate the mixed classes will be discussed later.) Then, the mixed classes, as well as other surface defects, are classified by using a weighted SVM. At the lower level, the surface defects in mixed classes are further classified by using separate classifiers, which are also weighted SVMs.

To discover the surface defects that are difficult to be distinguished, a weighted SVM classifier is initially used to classify all the surface defects. Then based on the confusion matrix of the classification result, an algorithm is developed to automatically generate the mixed classes. The confusion matrix for a n-class classification problem is traditionally defined as a $n \times n$ matrix **V** (Kohavi and Provost 1998). Each column of **V** represents the instances in a predicted class. $V_{i,j}$, the element of **V** at the i^{th} row and the j^{th} column, indicates the number of instances (classified as the j^{th} class) that truly belong to the i^{th} class. In order to represent the

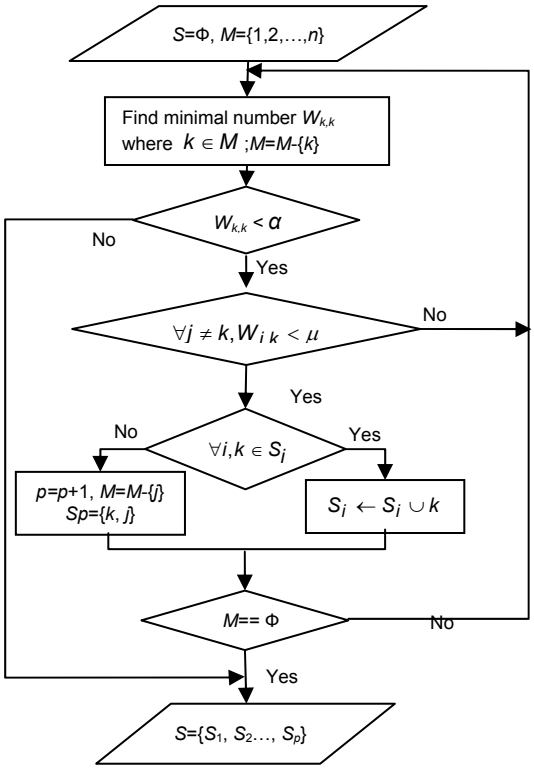

FIGURE 4. ALGORITHM 2 FOR GENERATING MIXED CLASSES.

classification accuracy for each class more intuitively, a matrix **W**, defined as confusion ratio matrix, is calculated based on confusion matrix **V. W** has the same dimension as **V** and the element of **W,** denoted as $W_{i,j}$**,** is calculated as follows:

$$W_{i,j} = \frac{V_{i,j}}{\sum_{j=1}^{n} V_{i,j}} \qquad (9)$$

where i,j = 1,2,…,n . $W_{i,j}$ is calculated as the ratio of $V_{i,j}$, the number of individuals in the i^{th} class that are classified as the j^{th} class, to $\sum_{j=1}^{n} V_{i,j}$, the total number of individuals in the i^{th} class. The i^{th} diagonal element of **W**, $W_{i,i}$, i=1,…,n represents the classification accuracy for the i^{th} surface defects. If $W_{i,i}$ is less than a threshold α, class i is not properly classified. To find the classes that have similar shapes to class i, each of $W_{i,j}$, $j \neq i$, is compared to a threshold μ. If $W_{i,j}$ is larger than μ, the instances in the i^{th} class are largely misclassified as the j^{th} class. As a result, the i^{th} class and the j^{th} class are difficult to be distinguished, and thus be merged as a mixed class. The detailed algorithm

to generate all the mixed classes $S=\{S_1,\ldots,S_{k,\ldots},S_p\}$, where S_k is the k^{th} mixed class and p is the number of mixed classes, is illustrated in Figure 4.

CASE STUDY

In this case study, the sensing images were collected at a random sampling time for several hours per day during three-months production at a real-world rolling process, in which the production line was operated under a wide range of working parameters to produce different products, such as carbon steel bars and alloy steel bars with the varying diameter sizes of [12, 80] mm; the reduction ratios of [0.00785 to 0.246], and the moving speed of the rolling bars of [1, 20] m/s, respectively. The temperature of the steel surface is about 800 ~ 1000° C. Based on the first step of the fast screening algorithm, 4463 suspicious defect regions are screened out from the tremendous sensing images. These suspicious images are then categorized and labeled as 10 classes based on the confirmed quality inspection, where the first class is the false positive and the other nine classes are the surface defects in the rolling process. Table 1 shows the ten types of classes and the corresponding sample sizes. The samples are randomly divided into the training data set and testing data set. For each class, the number of the samples assigned in the training data set is four times as that of the testing data set. The real labels of N samples are given as a benchmark for the evaluation of the classification method developed in this paper.

TABLE 1. CLASSES OF SURFACE DEFECTS.

Class Index	0	1	2	3	4
Defect type	positive false	Seam	Over fill	Crack	Lap
Sample size	192	608	113	112	705
Class Index	5	6	7	8	9
Defect type	Cross roll	Burn steel	Gauge	Rubbing	Pass wear
Sample size	38	262	286	500	1547

From feature generation process, 50 features are extracted for each sample image. These are used as the input of the feature selection.

Result of Feature Selection Based on GA and a Linear Classifier

Table 2 enlists the result of the feature selection based on GA and the linear classifier. From Table 2, we can see the number of selected features decreases from 50 to 31, while the classification accuracy increases from 54.83% to 77.84%.

TABLE 2. CLASSES OF SURFACE DEFECTS.

	Classification accuracy	Number of features
Feature pool	54.83%	50
Selected features	77.84%	31

Comparison of Classification Performance among Regular SVM, Weighted SVM, and Hierarchical Weighted SVM

Those 31 selected features are further used as the input of the classifiers. First, a regular SVM classifier is used, and the parameters in Equation 6 are listed as follows: C=2, and the RBF kernel for the regular SVM classifier is selected, i.e. $\phi(x_i)^T\phi(x_j)=\exp\left\{-\gamma\left\|x_i-x_j\right\|^2\right\}$, and $\gamma=1$.

When a regular SVM is applied, the overall classification accuracies are 91.89% and 85% for the training data and testing data, respectively. Tables 3 and 4 show the confusion ratio matrices for the testing data and training data, respectively.

TABLE 3. CONFUSION RATIO MATRIX OF THE TRAINING DATA USING A REGULAR SVM CLASSIFIER WITHOUT WEIGHTS AND A HIERARCHICAL STRUCTURE.

W	0	1	2	3	4	5	6	7	8	9
0	0.95	0.01	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.02
1	0.01	0.86	0.00	0.00	0.02	0.00	0.07	0.01	0.01	0.01
2	0.00	0.00	0.91	0.00	0.08	0.01	0.00	0.00	0.00	0.00
3	0.00	0.25	0.00	0.66	0.08	0.00	0.00	0.01	0.00	0.00
4	0.00	0.02	0.01	0.00	0.89	0.00	0.00	0.01	0.00	0.07
5	0.00	0.00	0.10	0.00	0.40	0.50	0.00	0.00	0.00	0.00
6	0.03	0.16	0.00	0.00	0.02	0.00	0.75	0.01	0.00	0.02
7	0.01	0.00	0.00	0.00	0.03	0.00	0.01	0.91	0.03	0.01
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.98	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99

From Tables 3 and 4, we can see that the classification accuracies of the sixth class (the fifth surface defects) for both the training data

and testing data (both are 50%) are much lower than those of the other classes. The reason lies in the fact that the sample size of the sixth class, 38, is much smaller than those of the other classes.

TABLE 4. CONFUSION RATIO MATRIX OF THE TESTING DATA USING A REGULAR SVM CLASSIFIER WITHOUT WEIGHTS AND A HIERARCHICAL STRUCTURE.

W	0	1	2	3	4	5	6	7	8	9
0	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.03
1	0.01	0.71	0.00	0.01	0.05	0.00	0.08	0.06	0.02	0.06
2	0.04	0.00	0.81	0.00	0.08	0.04	0.00	0.00	0.00	0.04
3	0.00	0.32	0.00	0.44	0.16	0.00	0.00	0.00	0.04	0.04
4	0.01	0.03	0.01	0.00	0.82	0.01	0.00	0.02	0.00	0.09
5	0.00	0.00	0.00	0.00	0.50	0.50	0.00	0.00	0.00	0.00
6	0.00	0.27	0.02	0.04	0.04	0.00	0.57	0.02	0.00	0.04
7	0.00	0.06	0.00	0.00	0.02	0.00	0.04	0.83	0.02	0.04
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.92	0.00
9	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97

In order to apply the weighted SVM formulated by Equation 7, the weights z_i are calculated based on Equation 8 and the sample size of each class represented in Table 2. z_i equals 11.7 if the i^{th} sample belongs to the sixth class, but otherwise, z_i equals 0.

TABLE 5. CONFUSION RATIO MATRIX OF THE TRAINING DATA USING A WEIGHTED SVM CLASSIFIER WITHOUT A HIERARCHICAL STRUCTURE.

W	0	1	2	3	4	5	6	7	8	9
0	0.95	0.01	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.02
1	0.01	0.86	0.00	0.00	0.02	0.00	0.07	0.01	0.01	0.01
2	0.00	0.00	0.86	0.00	0.05	0.09	0.00	0.00	0.00	0.00
3	0.00	0.25	0.00	0.66	0.08	0.00	0.00	0.01	0.00	0.00
4	0.00	0.02	0.01	0.00	0.86	0.02	0.00	0.01	0.00	0.07
5	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
6	0.03	0.16	0.00	0.00	0.02	0.00	0.75	0.01	0.00	0.02
7	0.01	0.00	0.00	0.00	0.03	0.00	0.01	0.91	0.03	0.01
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.98	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99

TABLE 6. CONFUSION RATIO MATRIX OF THE TESTING DATA USING A WEIGHTED SVM CLASSIFIER WITHOUT A HIERARCHICAL STRUCTURE.

W	0	1	2	3	4	5	6	7	8	9
0	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.03
1	0.01	0.71	0.00	0.01	0.05	0.00	0.08	0.06	0.02	0.06
2	0.04	0.00	0.81	0.00	0.08	0.04	0.00	0.00	0.00	0.04
3	0.00	0.32	0.00	0.44	0.16	0.00	0.00	0.00	0.04	0.04
4	0.01	0.03	0.01	0.00	0.80	0.04	0.00	0.02	0.00	0.09
5	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
6	0.00	0.27	0.02	0.04	0.04	0.00	0.57	0.02	0.00	0.04
7	0.00	0.06	0.00	0.00	0.02	0.00	0.04	0.83	0.02	0.04
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.92	0.00
9	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97

The overall classification accuracies when applying the weighted SVM are 91.80% and 85.11% for the training data and testing data, respectively. Tables 5 and 6 enlist the confusion ratio matrices for the testing and training data, respectively. Compared to the results of the regular SVM, the overall classification accuracies remain about the same for both the training data (91.80% compared to 91.89%) and the testing data (85.11% compared to 85.00%), while the classification accuracies for the sixth class increased significantly for both the training data (100% compared to 50%) and the testing data (100% compared to 50%).

TABLE 7. CONFUSION RATIO MATRIX OF THE TRAINING DATA USING A WEIGHTED HIERARCHICAL SVM CLASSIFIER.

W	0	1	2	3	4	5	6	7	8	9
0	0.95	0.01	0.00	0.00	0.00	0.00	0.01	0.02	0.00	0.02
1	0.01	0.87	0.00	0.00	0.03	0.00	0.05	0.01	0.01	0.01
2	0.00	0.00	0.86	0.00	0.05	0.09	0.00	0.00	0.00	0.00
3	0.00	0.18	0.00	0.77	0.03	0.00	0.00	0.01	0.00	0.00
4	0.00	0.01	0.01	0.00	0.87	0.02	0.00	0.01	0.00	0.07
5	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
6	0.03	0.16	0.00	0.00	0.01	0.00	0.75	0.01	0.00	0.02
7	0.01	0.00	0.00	0.00	0.03	0.00	0.01	0.91	0.03	0.01
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.98	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99

TABLE 8. CONFUSION RATIO MATRIX OF THE TESTING DATA USING A WEIGHTED HIERARCHICAL SVM CLASSIFIER.

W	0	1	2	3	4	5	6	7	8	9
0	0.95	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.03
1	0.01	0.73	0.00	0.01	0.02	0.00	0.08	0.06	0.02	0.06
2	0.04	0.00	0.81	0.00	0.08	0.04	0.00	0.00	0.00	0.04
3	0.00	0.28	0.00	0.56	0.08	0.00	0.00	0.00	0.04	0.04
4	0.01	0.01	0.01	0.00	0.81	0.04	0.01	0.02	0.00	0.09
5	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
6	0.00	0.18	0.02	0.00	0.00	0.00	0.73	0.02	0.00	0.04
7	0.00	0.06	0.00	0.00	0.02	0.00	0.04	0.83	0.02	0.04
8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.92	0.00
9	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97

Finally, the weighted hierarchical SVM is applied. Algorithm 2 is implemented with α being selected as 0.8 and μ being selected as 0.1. The result shows $S=\{S_1\}$ and $S_1=\{1,3,4,6\}$. That is, classes 1,3,4,6 are grouped as a single mixed class. At the higher level, a weighted SVM classifier is used to classify classes 0,2,5,7,8,9, and S_1; a separate weighted SVM is used at the lower level to further classify the classes in S_1, i.e., classes 1,3,4, and 6. Compared to the results of weighted SVM, the overall classification accuracies when applying the weighted hierarchical SVM increase for both the training data (increased from 91.80% to 92.34%) and the testing data (increased from 85.11% to 86.89%). Tables 7 and 8 show the confusion ratio matrices for the testing and training data

respectively. It can be seen that the classification accuracies for classes in mixed class S_1, i.e., classes 1, 3, 4 and 6 are increased for both the training data and the testing data, while the classification accuracies for other classes remain the same. Implementing the developed methods in an industrial test site showed that the speed satisfies the engineering specification for on-line classification of surface defects.

CONCLUSION

This paper proposes an effective method to online classify different types of surface defects in rolling processes. The method development includes three major steps: feature generation, feature selection, and classifier design. A new classifier, named weighted hierarchy SVM classifier, is proposed for improving the classification accuracy. A case study is conducted in a real-world rolling process, in which the classification performance of this newly designed classifier is compared to that of the regular SVM and the weighted SVM. The comparison results indicate that the proposed weighted hierarchy SVM outperforms the other two classifiers. It can improve not only the overall classification accuracy, buy also the classification accuracy for individual defects that have similar shapes or/and have small sample sizes. The application of the developed method to daily production shows that it can successfully classify the various surface defects with respect to different materials and operational conditions.

ACKNOWLEDGEMENTS

This research is partially supported by NSF SBIR F015886, NSF DMI F0541750, and Michigan 21st Century Jobs Fund F015886. The authors would like to thank Dr. Hongbin Jia at OG Technologies, Inc. for his invaluable help on this research.

REFERENCES

Cristianini, N., J.S. Taylor (2000). *An Introduction to Support Vector Machines.* Cambridge University Press, New York.

Duda, R.O., P.E. Hart, and D.G. Stork (2001). *Pattern Classification*, Wiley, New York.

Holland J.H. (1992). *Adaptation in Natural and Artificial System.* MIT Press, Cambridge.

Huang, Y.M., and S.X. Du (2005). "Weighted Support Vector Machine for Classification with Uneven Training Class Sizes." *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics,* Vol. 7, pp. 4365-4369.

Jin, N., S. Zhou, T.S. Chang and H.H.-H Huang (2008). “Identification of Influential Functional Process Variables for Surface Quality Control in Hot Rolling Processes.” *IEEE transactions on Automation Science and Engineering*, Vol. 5, pp. 557-562.

Kohavi, R., and F. Provost (1998). "Glossary of Terms." *Machine Learning*, Vol. 30, pp. 271-274.

Li, J., J. Shi, and T.S. Chang (2007). "On-Line Seam Detection in Rolling Processes Using Snake Projection and Discrete Wavelet Transform." *Journal of Manufacturing Science and Engineering*, Vol. 129, pp. 926-933.

Mitchell M. (1998). *An Introduction to Genetic Algorithms.* MIT Press, Cambridge.

Okamoto, Y., F. Kaminaga, and M. Osakabe (1988). "Detection of Surface Flaw by Infrared Radiation Sensor." *The 7th Sensor Symposium, Tokyo: Institution of Electricity.*

Sonka, M., V. Hlavac, and R. Boyle (1999). *Image Processing, Analysis, and Machine Vision, Second Edition*. Brooks/Cole, Thomson Asia Pte Led, Toronto, Canada.

Sugimoto, T., and T. Kawaguchi (1998). "Development of a Surface Defect Inspection System Using Radiant Light from Steel Products in a Hot Rolling Line." *IEEE Transactions, Instrumentation and Measurement,* Vol. 47, pp. 409-416.

Vapnik V. (1998). *Statistical Learning Theory.* Wiley, New York.

Ziou, D., and S. Tabbone, (1998). "Edge Detection Techniques-An Overview." *Pattern Recognition and Image Analysis*, Vol. 8, pp. 537-559.