AUTOMATIC MONITORING AND DIAGNOSIS OF THE DRESSING OPERATION THROUGH THE CLASSIFICATION OF TEXTURAL PATTERNS IN ACOUSTIC MAPS

Arthur Plínio de Souza Braga
Departamento de Ciências de Computação e Estatística,
Universidade de São Paulo,
Caixa Postal 668,13560-970, São Carlos, SP, Brazil
e-mail: arthurp@icmc.usp.br

André Carlos Ponce de Leon Ferreira de Carvalho
Departamento de Ciências de Computação e Estatística,
Universidade de São Paulo,
Caixa Postal 668,13560-970, São Carlos, SP, Brazil
e-mail: andre@icmc.usp.br

João Fernando Gomes de Oliveira
Departamento de Engenharia de Produção,
Universidade de São Paulo,
Av. Trabalhador Sãocarlense, 400,13566-590, São Carlos, SP, Brazil
e-mail: jfgo@sc.usp.br

Abstract. *A competitive manufacturing enterprise depends on a high level performance of its processes. In the metal-mechanic industry, such a requirement is pursued through the development of reliable monitoring systems. These systems must assure reliable information about the process itself, and about the machine's parameters. This paper proposes an efficient strategy for the automatic monitoring and diagnosis of dressing operations. The proposed system is based on Artificial Intelligence (AI) techniques, like neural networks, support vector machines, and decision trees, to classify textural features of an image, the acoustic map, which represents the interaction between the dresser and the grinding wheel. The classification indicates if the dressing operation should stop or not, what implies in a better use of the grinding wheel and costs reduction. The results obtained in the performed simulations are very promising, with 100% of right matches with the best tested classifiers. Such initial results point out to an increase in the production velocity, and the reducing in the number of defective pieces.*

Keywords*: Dressing Monitoring System, Acoustic Emission, Neural Networks, Decision Trees, Support Vector Machines.*

1. Introduction

Monitoring systems measure the conditions of a machine tool or of the process itself. Hence, monitoring must recognize defects and report them as a fault message or a system state (Maksoud and Atia, 2004). Some monitoring techniques simply give the measurements obtained from the sensors, some others make further analysis of these measurements in order to provide some other indirect parameters. The results obtained can be used as input for a diagnostic process in which the reasons for the fault are found and located. In the field of manufacturing processes, the main tasks of these systems are (Tönshoff et al., 2000): (i) Function control and fault location of machine components, (ii) Process control to recognize process failure, (iii) Recognition of machine inaccuracy, which leads to the lack of quality and (iv) Support of operating and maintenance staff. The focus of this work is on automatic fault diagnosis of the grinding wheel during dressing processes in sense to provide high product quality, associated to a reduction in the processing cost.

The dressing process prepares the grinding wheel, which has an important effect on grinding operations that usually determines the major portion of the processing cost (Tönshoff et al., 2000; Maksoud and Atia, 2004). Depending on the complexity of the manufacturing process, existing methods for process monitoring can be divided into: (i) Signal- or boundary-value-oriented methods; (ii) Model-based methods and (iii) Classifying methods. The particular nature of the grinding wheel, which contains many grains randomly spaced and placed within the periphery of the wheel, difficult the signal and the model-based solutions. Thus, the classifying methods try to find the link of a feature vector to a certain class of features. This vector is often determined by the extraction of relevant features from the process signals.

Here, the process signals are the Acoustic Emission (AE) generated during the interaction between the dresser and the grinding wheel. Li (2002) defines AE as*: "The class of phenomena whereby transient elastic waves are generated by the rapid release of energy from a localized source or sources within a material, or the transient elastic wave(s) so generated"*. The AE has been proven to contain information strongly related to the condition changes in the grinding zone (Kwak et al., 2004.a-b). The major advantage of using AE to monitor tool condition is that the frequency range of

the AE signal is much higher than that of the machine vibrations and environmental noises, and does not interfere with the cutting operation (Li, 2002).

Numerous studies have established the effectiveness of AE-based sensing methodologies for tool condition and cutting process monitoring (Tönshoff et al., 2000; Oliveira and Dornfeld, 2001; Li, 2002; Kwak and Ha, 2004.a-b; Maksoud and Atia, 2004). These AE-based monitoring systems generally use parameters like the peak value of the RMS, the peak value of a chosen frequency in a FFT output vector, the detection of the threshold, and the standard deviation of acquired AE signals (Kwak and Ha, 2004.a-b; Kwak and Song, 2001). However, two main limitations can be cited regarding these typical AE monitoring solutions (Oliveira and Dornfeld, 2001): (i) the oscillation of its RMS level and (ii) its saturation. Such limitations are, in general, reduced by adopting new signal processing techniques or multi sensor analysis – that can imply in additional costly computation.

An alternative solution for AE monitoring of a cutting process that can be very effective, and fast for the contact detection of moving surfaces, is the construction of an "acoustic map" of the grinding wheel. During the dressing operation, the interaction between dresser and grinding wheel can be acoustically mapped by using the experimental setup as the shown in Figure 1. The data acquisition is made in data arrays corresponding to a full rotation of the grinding wheel, and is triggered by a sensor positioned on the spindle. By using this approach, a more informative representation of the grinding wheel roughness is acquired.

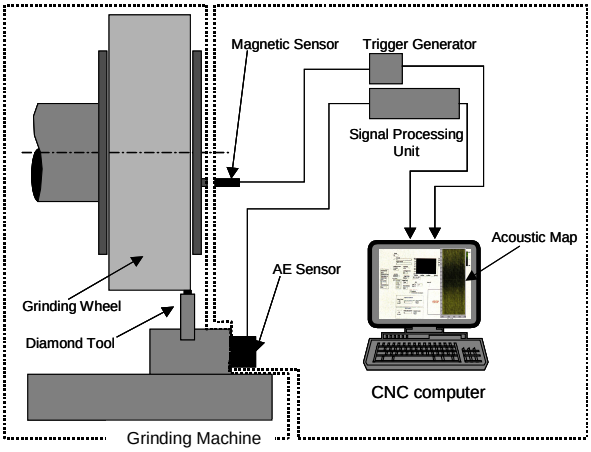


FIGURE 1 – AE mapping system (Oliveira et al., 2002).

An image, the acoustic map, is built up by representing the AE level of each acquired sample with a gray scale graph. During the dressing operation, the map is constructed in real time by adding columns in the array as the dressing tool travels along the wheel surface (Oliveira et al., 2002). The lack of contact between dressing tool and grinding wheel will appear as dark areas in the map. The vertical and horizontal directions are the wheel circumference and width respectively. The gray intensity shows the acoustic emission RMS value measured from the interaction between dressing tool and the abrasives grains. With acoustic maps, the limitations on AE-based monitoring is reduced, since this graph is not influenced by the RMS fluctuation along the time. The fluctuation would lead to changes in the image intensity or contrast of a given pattern. Therefore, the diagnosis based on the acoustic maps would not be influenced by temperature or other long-term AE disturbances (Oliveira and Dornfeld, 2001).

The innovative approach of Oliveira and Dornfeld (2001) allows that human operators to perform the monitoring of a dressing operation by simple visual inspection of the acoustic maps. The contribution of this work is in the automation of this monitoring and diagnosis through the use of Artificial Intelligence (AI) techniques (Winston, 1993; Haykin, 1994; Duda et al., 2000). Previous works have also adopted AI, in combination with AE measurements, a review can be found in (Li, 2002) and (Maksoud and Atia, 2004). However, these works used other parameters, like the before mentioned peak of RMS, the peak of FFT (Fast Fourier Transform), the detection of the threshold, and the standard deviation of acquired AE signals instead of an image, like the acoustic maps (more complex patterns).

Based on techniques from image processing (Kulkarni, 1994; Egmont-Petersen et al., 2002), the monitoring system proposed on Section 2 can be seemed as a classifying method where Haralick textural descriptors (Haralick et al., 1973) are used as features to represent the acoustic map patterns. In fact, only a subset of the Haralick descriptors is shown to be enough to describe the clusters of the current problem. Real acoustic maps, collected from a CNC grinding machine at the Laboratory for Optimization of Manufacturing Processes (OPF), are used in our experiments to test our monitoring system. The results (Section 3) are very promising and point out to a sophisticated monitoring system able to either replace or provide support to skilled operators.

This paper is organized as follows: Section 2 presents the proposed monitoring and diagnosis system of dressing operations - the extracted textural features are detailed, and some principles of the adopted classifiers are discussed. Section 3 describes the experimental setup and shows the obtained results. A general discussion on the obtained results and final conclusions are presented in Section 4.

2. Automatic Monitoring and Diagnosis of Dressing Processes

The success of a pattern classifier depends on the chosen of a proper (i) representation and (ii) algorithm for the specific problem. The next subsections discuss these issues.

2.1. Image representation

Texture is one of the most important characteristics used to classify regions of interest in an image (Haralick et al., 1973; Kulkarni, 1994). Intuitively, texture descriptors provide measures of properties, such as smoothness, coarseness, and regularity. A statistical algorithm proposed by Haralick and colegues (1973), and based on gray level co-occurrence matrices (GLCM) to extract the measures, is used in this work to extract the textural features of the acoustic maps. In statistical methods, features are described using a spatial gray level dependency matrix. For a two-dimentional image $f(x,y)$ with N discrete values, the spatial gray level dependency matrix $\mathbf{P}(d,\phi)$ is described for each d and ϕ. The element $p_{i,j}$ is defined as the relative number of times a gray level pair (i,j) occurs when pixels separated by the distance d along the angle ϕ are compared. Each element is finally normalized by the total number of occurrences giving the co-occurrence matrix $\mathbf{P}$. A spatial gray level dependency matrix is also called a co-occurrence matrix. It is given by:

$$\mathbf{P}(d,\phi)=\begin{bmatrix} p_{0,0} & p_{0,1} & \cdots & p_{0,N-1} \\ p_{1,0} & p_{1,1} & \cdots & p_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ p_{N-1,0} & p_{N-1,1} & \cdots & p_{N-1,N-1} \end{bmatrix} \tag{1}$$

where $p_{i,j}$ is given by (Kulkarni, 1994):

$$p_{i,j}=\frac{\text{number of pixel pairs with intensity } (i,\ j)}{\text{total number of pairs considered}} \tag{2}$$

The number of operations required to process the matrix is directly proportional to the number of resolution cells N present in the image. In comparison, the number of operations are of the order $N.\log N$ if one wishes to use Fourier or Hadamard transform to extract texture information (Haralick et al., 1973). Besides, to compute the entries in the gray-tone spatial-dependence matrices, one needs to keep only two lines of image data in core at a time. Thus, no severe storage constraints are imposed (Haralick et al., 1973). The assumption in characterizing image texture is that all the texture information is contained in the gray-tone spatial-dependence matrices. Hence, all the textural features proposed by Haralick (Haralick et al., 1973) are extracted from these matrices. In Haralick's original papers, 14 measurements are described. However, only six descriptors are adopted in this work:

1) Angular Second Moment: $$f_1 = \sum_i \sum_j P(i,\ j)^2_{\Delta x,\Delta y} \tag{3}$$

2) Contrast: $$f_2 = \sum_i \sum_j (i-j)^2 P(i,\ j)_{\Delta x,\Delta y} \tag{4}$$

3) Correlation: $$f_3 = \frac{\sum_i \sum_j i \cdot j \cdot P(i,\ j)_{\Delta x,\Delta y} - \mu_l \cdot \mu_c}{\sigma_l \cdot \sigma_c} \tag{5}$$

4) Sum of Squares (Variance): $$f_4 = \sum_k k P^D(k)_{\Delta x,\Delta y} \tag{6}$$

5) Inverse Difference Moment: $$f_5 = \sum_i \sum_j \frac{P(i,\ j)_{\Delta x,\Delta y}}{1+(i-j)^2} \tag{7}$$

6) Sum Average: $$f_6 = \sum_k k P^S(k)_{\Delta x,\Delta y} \tag{8}$$

where:

$$\mu_l = \sum_i i \cdot P(i,*)_{\Delta x,\Delta y}\ ,\ \mu_c = \sum_j j \cdot P(*,\ j)_{\Delta x,\Delta y}\ ,\ \sigma_l = \sqrt{\sum_i (i-\mu_l)^2 \cdot P(i,*)_{\Delta x,\Delta y}}\ ,$$

$$\sigma_c = \sqrt{\sum_j (j-\mu_c)^2 \cdot P(*,\ j)_{\Delta x,\Delta y}}\ ,\ P(i,*)_{\Delta x,\Delta y} = \sum_j P(i,\ j)_{\Delta x,\Delta y}\ ,\ P(*,\ j)_{\Delta x,\Delta y} = \sum_i P(i,\ j)_{\Delta x,\Delta y}$$

These features characterize texture patterns. Based on these measures, a vector representation of an acoustic map, $[f_1, f_2, f_3, f_4, f_5, f_6]$, can be created.

2.2. Pattern Classification

The problem now is to obtain a partition of the Haralick textural descriptors feature space into different regions, each one associated to a different decision - dressing or not dressing the grinding wheel. In our work, pattern classification (Duda et al., 2000) is the way to obtain the decision boundary between these patterns, through a classifier that maps a feature vector into a label related to the action to be performed. A classification task usually involves trainings and testing data that consist of some data instances. Each instance in the training set contains one "target value" (class labels) and several "attributes" (features). One central aim of a classifier is to suggest actions even when presented to novel patterns (Duda et al., 2000) - this is the issue of generalization. In dressing monitoring, this characteristic is particularly important, since it is impossible to collect beforehand all possible acoustic maps to design our classifier. So, four classifiers, with learning[1] and generalization capabilities, were used in our work:

(i) Multi-Layer Perceptron (MLP): A MLP network is made up of several layers of neurons (beyond input and output layers, some neurons are organized in intermediate, or hidden, layers). Each layer is usually fully connected to the next one. The input signal is propagated through the network in a forward direction, on a layer-by-layer basis. MLP networks are universal approximators, i.e. they can be viewed as a nonlinear input-output of a general nature (Haykin, 1994). MLPs learn a mapping by supervised training, feeding input-output examples to a highly popular algorithm known as *error back-propagation*. The back-propagation is a specific technique for implementing gradient descent in weight space. The basic idea is to efficiently compute partial derivatives of an approximating function *F(*w;x*)* - the input-output mapping - with respect to all the elements of the adjustable weight matrix **w** for a given value of input vector x. The output of neuron *j* in layer *l* is given by (Haykin, 1994):

$$y_j^{(l)}(n) = g\left(\sum_{i=0}^{p} w_{ji}^{(l)}(n) \cdot y_i^{(l-1)}(n)\right) \tag{9}$$

where $y_i^{(l-1)}(n)$ is the output of neuron *i* in the previous layer *l-1* at iteration *n*, $w_{ji}^{(l)}(n)$ is the synaptic weight of neuron *j* in layer *l* that is fed from neuron *n* in layer *l-1*, and *g(.)* is the activation function. In *Backward step*, all the weights are adjusted in accordance with the error back-propagation algorithm. Given the desired network response vector **d**(*n*), each neuron *j* in the output layer should reach response $d_j(n)$. Thus, these adjusts are made through the computation of the local gradients

$$\delta_j^{(L)}(n) = y_j^{(L)}(n) \cdot \left[1 - y_j^{(L)}(n)\right] \cdot \left[d_j - y_j^{(L)}(n)\right] \qquad \text{for neuron } j \text{ in output layer } L \tag{10}$$

$$\delta_j^{(l)}(n) = y_j^{(l)}(n) \cdot \left[1 - y_j^{(l)}(n)\right] \cdot \sum_k \delta_j^{(l+1)}(n) \cdot w_{kj}^{(l+1)}(n) \qquad \text{for neuron } j \text{ in hidden layer } l \tag{11}$$

Hence, the network synaptic weights update in layer *l* is made according to the following generalized delta rule:

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha\left[w_{ji}^{(l)}(n) - w_{ji}^{(l)}(n-1)\right] + \eta\delta_j^{(l)}(n) y_i^{(l-1)}(n) \tag{12}$$

where η is the learning-rate parameter and α is the momentum constant.

(ii) Radial-Basis Function (RBF): In RBF networks, the learning of the mapping between the feature vector and the actions can be carried out in different ways. While this mapping in MLPs is based on optimization methods known in statistics as *stochastic approximation* (Haykin, 1994), RBF networks take the mapping as a *curve-fitting (approximation) problem* in a high-dimensional space. The RBF technique consists of choosing a function *F* that has the following form (Powell, 1988):

$$F(x) = \sum_{i=1}^{N} w_i \varphi\left(\left\| x - x_i \right\|\right) \tag{13}$$

where: $\{\varphi(\|x-x_i\|),\ i = 1,2,...,N\}$ is a set of *N* arbitrary (generally nonlinear) functions, known as *radial-basis functions*, and $\|.\|$ denotes a *norm* that is usually taken to be Euclidean (Haykin, 1994). The principle for the approximation of desired mapping *F(*x*)*, Eq. (13), is to solve, for the unknown weights $\{w_i\}$ - vector **w**, the following set of simultaneous linear equations (Haykin, 1994):

[1] Learning refers to some form of algorithm for reducing the error on a set of training data. In the broadest sense, any method that incorporates information from training samples in the design of a classifier employs learning (Duda et al., 2000).

$$\underbrace{\begin{bmatrix} \varphi_{11} & \varphi_{12} & \cdots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \cdots & \varphi_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ \varphi_{N1} & \varphi_{N2} & \cdots & \varphi_{NN} \end{bmatrix}}_{\mathbf{\Phi}} \cdot \underbrace{\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}}_{\mathbf{w}} = \underbrace{\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix}}_{\mathbf{d}} \tag{14}$$

where: Φ is the *interpolation matrix* and **d** is the desired response vector. Haykin (1994) argues that Φ is positive definite, and cites (Miccheli, 1986) to justify that the interpolation matrix is nonsingular - so, we can solve (14) by:

$$\mathbf{w} = \Phi^{-1} \cdot \mathbf{d} \tag{15}$$

where Φ^{-1} is the inverse of the interpolation matrix Φ.

(iii) Support Vector Machines (SVMs): SVM represents a new technique for data classification (Hsu et al., 2003). Given a training set of instance-label pairs (x_i, y_i), a SVM computes the solution through an optimization problem that finds a linear hyperplane $\mathbf{w} \cdot \mathbf{x}_i + b$ able of separating data with a maximal margin. The recent interest in SVMs is motivated by (Smola et al., 1999): (i) great generalization capability; (ii) robust in high dimensions, example: images; (iii) convex goal function; as a SVM solution implies in the optimization of a quadratic function, there is only one global minimum - what represents a remarkable advantage in comparison with neural networks with multiple global minimums; (iv) a solid theoretical background based on learning statistics theory (Vapnik, 1995). This is accomplish by the following optimization problem (Lorena and Carvalho, 2003):

$$\textbf{Minimize: } \|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \varepsilon_i \text{ , with } \textbf{Restrictions: } \begin{cases} \varepsilon_i \geq 0 \\ y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \varepsilon_i \end{cases} \tag{16}$$

where $\mathbf{x}_i \in \Re^m$, C is a constant that imposes a tradeoff between training error and generalization and the ε_i are the slack variables. The decision frontier obtained is given by (Lorena and Carvalho, 2003):

$$F(\mathbf{x}) = \sum_{\mathbf{x}_i \in SV} y_i \alpha_i \mathbf{x}_i \cdot \mathbf{x} + b \tag{17}$$

where α_i are Lagrange multiplies determined in the optimization process, and *SV* corresponds to the set of support vectors, patterns for which the associated Lagrange multipliers are larger than zero.

(iv) Decision trees (DTs): In general, a DT is an arrangement of tests that prescribes the most appropriate test at every step in the tree construction (Quinlan, 1986). Each decision outcome, a split, corresponds to splitting a subset of the training data. Each node is connected to a set of possible answers. Each nonleaf node is connected to a test that splits its set of possible answers into subsets corresponding to different test results. Each branch carries a particular test result's subset to another node. In general, DTs represent a disjunction of conjunctions of constraints on the attribute-values of instances. Each path from the tree root to a leaf corresponds to a conjunction of attribute tests, and the tree itself to a disjunction of these conjunctions (Winston, 1993).
There are several implementations of DTs, like CART, ID3 and C4.5 (see Breiman et al., 1984 and Mitchell, 1997 for a review), but the fundamental principle underlying tree creation is that of simplicity: the tree should be simple and compact tree with few nodes. In this work, one of the first and best-known tree creation algorithm, the CART algorithm (Breiman et al., 1984), is adopted for implementation. It uses a property test at each node *n* that makes the data reaching the immediate descendent nodes as pure as possible. In formalizing this notion, it turns out to be more convenient to define the impurity, rather than the purity of a node, by the entropy impurity measure (Duda et al., 2000):

$$i(n) = -\sum_j P(c_j).\log_2 P(c_j) \tag{18}$$

where *i(n)* denote the impurity of anode *n*; $P(c_j)$ is the fraction of patterns at node *n* that are in category c_j. By the well-known properties of entropy, if all the patterns are of the same category, the impurity is 0; otherwise it is positive, with the greatest value occurring when the different classes are equally likely.

The previous classifiers described were employed in the experiments performed. The main goal of the experiments was the separation, in the Haralick's descriptors space, of the maps associated with grinding wheels. Two classes were used: maps indicating that the wheel needs to be re-dressed, and maps that indicate good dressing. Next section comments the experimental setup and shows the obtained results.

3. Experiments and Results

The acoustic maps were acquired on-line using a CNC open architecture from the Laboratory for Manufacturing Processes Optimizing Group (OPF) of the University of São Paulo (USP). The CNC platform is based on the GB-FANUC platform 180i dual processor and a Pentium IV based interface.

The computation of (i) the co-occurrence matrix, (ii) the textural descriptors and (iii) the classifiers output were performed off-line using MATLAB® routines developed by the authors. In this work, the MLP network was a 6-6-1 one-hidden-layer architecture, with learning rate of 0.08; the RBF network used 10 Gaussian 0.001spread radial basis functions; the SVM used parameter C equal to 10 and the DT's impure nodes must have 5 or more observations to be split. The following results present an analysis of classifiers performance.

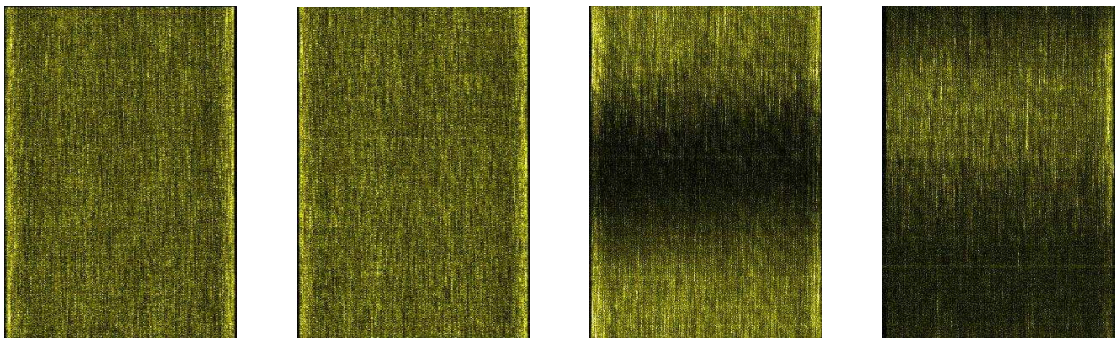
3.1. Experimental setup

The signal processing, the monitoring of the acoustic maps and the execution of controlling routines were performed through a graphical interface developed by OPF in LabView®. The experimental conditions used in collecting the acoustic maps are listed in Tab 1.

Table 1. Experimental specifications and conditions.

Items	Specification and conditions
Grinding wheel	SGB 60 L VHB
Grinding wheel velocity	1770 rpm
Depth	0.02 mm
Grinding wheel width	54 mm
Dressing speed	200 mm/min

The data were collected using a real grinding wheel to generate the acoustic maps. The data set was divided into two classes: (i) balanced grinding wheel (Fig. 4.a-b) and (ii) non-balanced grinding wheel (Fig. 4.c-d). The balance of the grinding wheel was controlled by changing the setup of the CNC control program.



(a) (b) (c) (d)

Figure 4. Example of acoustic maps: (a)-(b) stop dressing, (c)-(d) continue dressing.

For the training of the classifiers, 20 patterns from each class were used. A different set of patterns, with the same distribution of 50% stop dressing patterns and 50% of continue dressing patterns, was used in the test phase.

3.2. Simulation results

In the training phase, the maps set was randomly presented to the classifiers. Images were classified in two groups related with the balance of the wheel grinding: 0 (balanced wheel and no need of additional dressing operations) and 1 (non-balanced wheel and the need of more dressing operations).

Table 2 summarizes the obtained MSE results for the training and test set. One difference should be noted in these results: the outputs of the MLP and the RBF networks are continuous values, while the outputs of the SVM and the decision tree are discrete values (in this case, either 0 or 1). With this observation in mind, one can see three main aspects in the results shown in Table 2: (i) the outputs of the MLP and the RBF networks are very close to the desired values; (ii) only the SVM presented some misclassification of the maps (3 cases), and (iii) the DT performed a perfect match for all the tested patterns. From the fact that the MLP and RBF networks have continuous outputs, the results in Table 2 place, in a first view, the MLP, the RBF and the decision tree as very promising algorithms for the purpose of monitoring the dressing operation, since all presented 100% of correct classifications.

Table 2. Training set – Measures of the squared error.

	Training set				Test set			
squared error	MLP	RBF	SVM	DT	MLP	RBF	SVM	DT
Mean	0.0002	0.0036	0.0750	0.0000	0.0002	0.0138	0.0000	0.0000
Deviation	0.0003	0.0069	0.2667	0.0000	0.0009	0.0209	0.0000	0.0000
Max	0.0013	0.0335	1.0000	0.0000	0.0060	0.0934	0.0000	0.0000
Min	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Accuracy	100%	100%	92.5%	100%	100%	100%	100%	100%

The observed results for the SVM show 3 misclassification cases, what implies in a performance of 92.5% of correct classifications (Table 2) – a reasonable result that can be due to a bad choice of the SVM's parameters, since the misclassified maps have characteristics that are very particular to their class, as one can see in Figure 5. After the training phase, a set of different maps, the test set, was used to confirm that the classifiers learned the difference between balanced and non-balanced grinding wheels. Table 2 resumes these results.

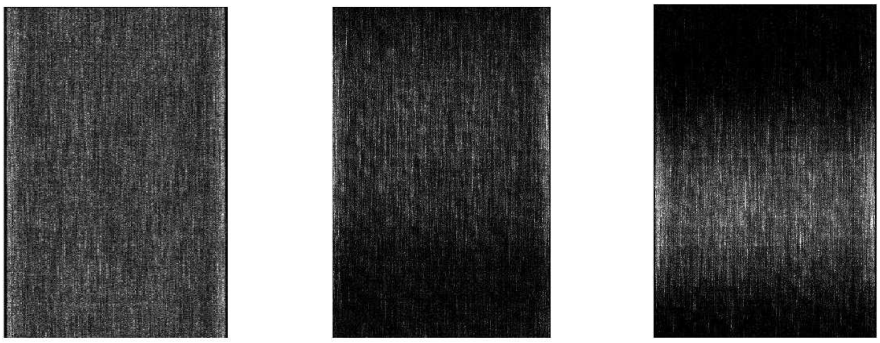


Figure 5. The three misclassified maps by SVM in the training set.

Generalization is an important aspect for a classifier working in a monitoring system. Thus, to obtain better estimatives of the generalization performance of the classifiers, the 80 collected acoustic maps were divided in training and test sets following the 5-fold cross-validation methodology (Mitchell, 1997). According to this method, the dataset is divided in five disjoint subsets of equal size. In each train/test round, four subsets are used for training and the remaining is left for test. This makes a total of five pairs of training and test sets. The error and the accuracy of a classifier on the total acoustic map dataset are then given by the average of the squared errors (Table 3) and accuracies (Table 4) observed in each test partition.

Table 3. Measures of the average squared error using 5-fold cross-validation.

Stage	MLP	RBF	SVM	DT
Training	0.0001 ± 0.0000	0.0042 ± 0.0016	0.0000 ± 0.0000	0.0031 ± 0.0070
Test	0.0118 ± 0.0212	0.0094 ± 0.0077	0.0625 ± 0.0765	0.0250 ± 0.0342

Table 4. Accuracy average using 5-fold cross-validation.

Stage	MLP	RBF	SVM	DT
Training	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	99.4 ± 1.4
Test	97.5 ± 5.6	100.0 ± 0.0	87.5 ± 15.3	95.0 ± 6.8

The four classifiers presented good generalization: all of them had a performance above of 87.5% of correct classifications for the test set patterns (Table 4). This promising performance again points out that the combination of acoustic map, textural descriptors and Artificial Intelligence techniques can be a simple and viable alternative for the implementation of an automatic monitoring and diagnosis system of dressing operations.

4. Conclusions

A reliable diagnosis of the grinding wheel is a relevant tool in metal-mechanic industry since it can contribute to costs reduction. However, the creation of a grinding wheel analytical model with its randomly distributed abrasive grains is not a trivial task. Thus, an approach to perform automatic monitoring and diagnosis of dressing operations is

the adoption of a classify method. In this paper, a ML-based approach for establishing an intelligent monitoring and diagnosis system of dressing processes has been proposed. Instead of sensor fusion solutions, or Fourier spectrum analysis of the AE signals, the proposed solution is based on: (i) acoustic maps which are not influenced by the RMS fluctuation along the time and (ii) pattern classification of textural descriptors of these maps through AI classifies, which involves less operations. Such an original combination was tested using four different classifiers: a MLP neural network, a RBF neural network, a Support Vector Machine and a Decision Tree. The performed experiments suggest that this strategy is very promising for future implementations of a monitoring system. In the current results, some minor result differences were observed that could be caused by the algorithms' adopted parameters.

5. Acknowledgements

The authors would like to thank the support from FAPESP and CNPq. The Laboratory for Optimization of Manufacturing Processes (OPF) provided the facilities to perform the experiments. Thanks also to Dr. Ana Patrocinio for her helpful comments regarding the Haralick's descriptors.

6. References

Breiman,L.; Friedman, J. H.; Olshen, R. A.; Stone, C. J., 1984, "Classification and Regression Trees", Wadsworth &Brooks/Cole Advanced Books & Software.

Duda, R. O.; Hart, P. E.; Stork, D. G., 2000, "Pattern Classification", 2nd Edition, Wiley-Interscience.

Egmont-Petersen, M.; Ridder, D.; Handels, H., 2002, "Image Processing with Neural Networks – A Review", Pattern Recognition, Vol. 35, pp. 2279-2301.

Haralick, R. M.; Shanmugam, K.; Dinstein, I., 1973, "Textural Features for Image Classification". IEEE Trans. Syst., Man, Cyber., Vol. SMC-3, pp. 610-621.

Haykin, S., 1994, "Neural Networks – a comprehensive foundation". Macmillan College Publishing Company, Inc.

Hsu, C. -W.; Chang, C.-C.; Lin, C.-J., 2003, "A Practical Guide to Support Vector Classification". (http://www.csie.ntu.edu.tw/~cjlin/papers.html).

Kulkarni, A. D., 1994, "Artificial Neural Networks for Image Understanding", Van Nostrand Reinhold – an International Thomson Publishing Company.

Kwak, J. –S.; Song, J. –B., 2001, "Trouble diagnosis of the grinding process by using acoustic emission signals". International Journal of Machine Tools & Manufacture, Vol. 41, pp. 899-913.

Kwak, J. –S.; Ha, M. –K., 2004.a, "Neural Network Approach for Diagnosis of Grinding Operation by Acoustic Emission and Power Signals". Journal of Materials Processing Technology, Vol. 147, pp. 65-71.

Kwak, J. –S.; Ha, M. –K., 2004.b, "Intelligent Diagnostic Technique of Machining State for Grinding". Int. J Adv. Manuf. Technol. Vol. 23, pp. 436-443.

Li, X., 2002, "A Brief Review: Acoustic Emission Method for Tool Wear Monitoring During Turning". International Journal of Machine Tools & Manufacture, Vol. 42, pp. 157-165.

Lorena, A. C. & Carvalho, A.C.P.L.F., 2003, "Introdução à Máquinas de Vetores Suporte". Relatório Técnico do Instituto de Ciências Matemáticas e de Computação (USP/São Carlos) No. 192.

Maksoud, T. M. A.; Atia, M. R., 2004, "Review of Intelligent Grinding and Dressing Operations". Machining Science and Technology, Vol. 8, No. 2, pp. 263 - 276.

Miccheli, C. A., 1986, "Interpolation of scattered data: distances matrices and conditionally positive definite functions". Constructive Approximation, V.2, pp: 11-22.

Mitchell, T. M., 1997, "Machine Learning". The McGraw-Hill Companies, Inc.

Oliveira, J. F. G.; Dornfeld, D. A., 2001, "Application of AE Contact Sensing in Reliable Grinding Monitoring". Annals of Cirp 2001, Berne, Swisserland, Vol. 51, No. 1, pp. 217-220.

Oliveira, J. F. G.; Silva, Eraldo Jannone da; Biffi, Marcelo et al., 2002, "New Architecture Control System for an Intelligent High Speed Grinder". Abrasives Magazine, USA, Vol. 2002, No. 06, pp. 04-11.

Powell, M.J.D., 1988, "Radial basis function approximations to polynomials". Numerical Analysis 1987 Proceedings, pp: 223-241, Dundee, UK.

Quinlan, J.R., 1986, "Induction of Decision Trees", Machine Learning, Vol. 1, pp. 81-106.

Smola, A.J.; Barlett, P.; Schölkopf, B. & Schurmmans, D., 1999, "Advances in Large Margin Classifiers". MIT Press.

Tönshoff, H. K.; Jung, M.; Männel, S.; Rietz, W., 2000, "Using Acoustic Signals for Monitoring Production Processes". Ultrasonics, Vol. 37, pp. 681-686.

Vapnik, V. N. "The Nature of Statistical Learning Theory". Springer-Verlag, 1995.

Winston, P. H., 1993, "Artificial Intelligence". Addison-Wesley.

7. Responsibility notice

The authors are the only responsible for the printed material included in this paper.