Automated Operating Mode Classification for Online Monitoring Systems

Article *in* Journal of Vibration and Acoustics · August 2009

DOI: 10.1115/1.3142871

M. G. Lipsett
University of Alberta

Jordan Mcbain
Dogged Mechatronics

Chris Mechefske
Queen's University

Markus A. Timusk
Laurentian University,
Sudbury, ON, P3E 2C6, Canada
e-mail: mtimusk@laurentian.ca

Michael G. Lipsett
University of Alberta,
Edmonton, AL, T6G 2G8, Canada
e-mail: michael.lipsett@ualberta.ca

Jordan McBain
Laurentian University,
Sudbury, ON, P3E 2C6, Canada

Chris K. Mechefske[1]
Queen's University,
Kingston, ON, K7L 3N6, Canada
e-mail: chrism@me.queensu.ca

Automated Operating Mode Classification for Online Monitoring Systems

Transient operation of machinery can greatly complicate the task of vibration-based online condition monitoring. Because the operating mode of a machine affects the physical response and hence the diagnostic parameters, real-time information regarding the operating mode is likely to improve the performance of an online fault detection system. This paper proposes a method for automated operating mode classification to augment the performance of vibration-based online condition monitoring systems for applications such as gearboxes, motors, and their constituent components. Experimental work has been carried out on the swing machinery of an electromechanical excavator, which demonstrates how such a method might function on actual dynamic signals gathered from an operating machine. Several variations of the system have been tested and found to be successful. [DOI: 10.1115/1.3142871]

1 Introduction

Machine condition monitoring can be defined as two complementary tasks. The first task characterizes the response of a machine during normal fault-free operations using diagnostic indicators (from measurements such as vibration and temperature). The second task compares diagnostic indicators from the same machine, during a period of unknown condition, to the original measurements. It is the deviation of the diagnostic indicators representing the unknown condition from the indicators representing the known condition that provides the estimate of the true condition of the machine. The accuracy of a condition estimate depends on a number of factors; the most significant being the sensitivity of the feature that is being observed to a given failure mode. While it is desirable for the diagnostic features to be sensitive to machine condition, it is also imperative that these features are insensitive to factors that are not caused by the machine condition. One such factor is the operating mode of the machine.

In many cases, the sensitivity of the diagnostic indicators to operating mode is not a concern. The most obvious example is for those machines that always operate in the same well-defined mode of operation (a familiar example of such a machine is a pump that runs at a steady speed). There is a category of machines whose operation can be described as continuously changing in duty (such as load and speed). This type includes machines that perform a set of repeatable but variable tasks, often controlled by a human operator. One example of such a machine is an excavator performing a continuous digging task. These machines are often overlooked as candidates for online condition monitoring because of the mode-related influences on the diagnostic parameters. It is this category of machines that is the focus of this research.

2 Approaches to the Problem

A number of techniques have been developed to mitigate the effects of varying operating mode on diagnostic indicators used for online condition monitoring. These methods are outlined in Secs. 2.1–2.4.

2.1 Signal Processing Approach. One of the most common approaches to condition monitoring of non-steady-state systems removes the variation caused by mode, by extracting the mode-independent signal components at the signal processing level. For example, in condition monitoring of jet engine vibration spectra, where the only data available have been collected during start up and shut down transients, Hayton et al. [1] used an order tracking technique to extract individual orders and reduce the effects of speed variation on frequency domain features. The removal of the influence of load fluctuations on the rotating equipment has been proven to be more difficult. For example, Baydar and Ball [2] and Chen et al. [3] used a time-frequency distribution called instantaneous power spectrum (IPS), which detects faults in machinery operating under different loading conditions. Zhan et al. [4] used a time-varying autoregressive modeling technique to extract load insensitive features from gear vibration signals.

2.2 Large Normal Condition Model. The approach termed as large normal condition modeling involves collecting data over the entire range of machine operating conditions and then building a single reference model of fault-free operation, which encompasses all operating conditions [5]. The difficulty with this approach is that two different, and often mutually exclusive, objectives have to be satisfied: The modeling technique needs to be general enough to recognize the entire class of normal operations, while still maintaining the sensitivity required to detect fault conditions.

2.3 Model Normalization Approach. Data normalization is a procedure to adjust data sets before they are introduced for classification. Signal changes that are known to be caused by variations in operating mode or environmental influences can thus be separated from those relating to condition. This approach has been employed by Sohn et al. [6] for monitoring the vibrations of structures, using neural network-based novelty detectors to detect faults. This approach appears to be most applicable to systems, which has a single, easily identifiable, dominant influencing parameter and where the changes due to this parameter can be easily predicted, modeled (parameterized), and reproduced.

2.4 Mode-Specific Diagnostic Routine Approach. A fourth strategy for online fault detection of unsteady machinery involves identification of the machine's current operating mode (using

[1]Corresponding author. Also at Department of Mechanical and Materials Engineering, McLaughlin Hall, Queen's University, Kingston, ON, K7L 3N6, Canada.

Contributed by the Technical Committee on Vibration and Sound of ASME for publication in the JOURNAL OF VIBRATION AND ACOUSTICS. Manuscript received May 9, 2007; final manuscript received April 15, 2009; published online June 5, 2009. Assoc. Editor: Jean Zu. Paper presented at the ASME 2007 Design Engineering Technical Conferences and Computers and Information in Engineering Conference (DETC2007), Las Vegas, NV, Sept. 4–7, 2007.

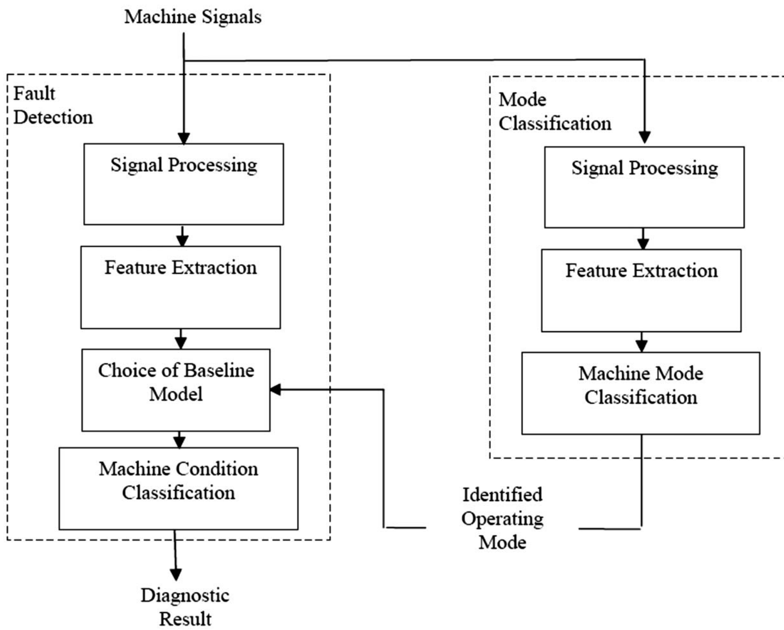


Fig. 1 Block diagram of signal processing system

some manual or automatic method) and using this information to put machinery measurements into context. This way, a specialized reference model, corresponding to an identified operating mode, is used for the assessment of the condition.

Real-time knowledge of the operating mode can assist the fault detection system in a number of ways. A choice of appropriate reference model can be made that reflects the specific operating mode. A specification can be defined for an appropriate tolerance for deviation from a baseline, e.g., some modes of operation may be more tightly defined than others. Alternatively, a decision can be made as to whether the present mode of operation is even suitable for monitoring, e.g., the machine is performing an unusual task. The general dataflow for this type of approach is depicted in Fig. 1.

This approach has been used by Timusk et al. [7] for the monitoring of a variable speed hydraulic pump. In that system, multiple specialized classifiers have been trained and dispatched, corresponding to the status of a single dominant parameter (the speed of a hydraulic pump). Performed by a basic expert system, the mode classification subsystem has been found to greatly improve the performance of the fault detection program.

The utility of an online indication of operation mode is further confirmed by the emergence of commercial condition monitoring systems, which can take advantage of the results of a preliminary mode classification. For example, SKF Corporation has recognized the need for mode-sensitive monitoring systems and has introduced a condition monitoring system that accepts as an input to its measurement and alarm capabilities the ability to control measurement evaluation by operating mode. This is done via a binary input to the monitoring system. The binary inputs are used to control an expert rule base, which is configured by the user [8].

3 Experiments

The industrial application with which this research is concerned is the swing machinery of an electromechanical excavator shown in Fig. 2, a machine that executes continually non-steady-state tasks, which are characterized by varying speeds under either fully loaded or unloaded conditions. The swing machinery is a subsystem that consists of two parallel motor-gearbox assemblies driving a common ring gear. The planetary-reduction type gearbox and motor assembly is shown in Fig. 3. The function of the swing machinery is to drive the shovel about its vertical axis as material is dug and deposited into waiting haul trucks. In performing this task, the gearbox is subjected to a constantly changing, but repeatable, operating mode.

In rotating machinery, the kinematics of the mechanical components form a relation between the rotating speed and characteristic frequencies, such as those of gear meshing and ball-bearing rotation. This relation is shown clearly in the vibration spectrogram of the swing machinery in Fig. 4, in which the gear meshing orders clearly follow the speed (shown on the lower half of Fig. 4). Because of this behavior, the mode of operation has a direct influence on the diagnostic indicators, and continuous online condition monitoring has not been effective. For this reason, the current approach to vibration monitoring of unsteadily operating equipment is to take it offline (out of production) and run it in a steady fashion as possible, with respect to how the machine has been operated when baseline measurements have been taken, to reduce the influence of speed and load variability.

Further examination of the vibratory response depicted in Fig. 4 also reveals that there are repetitive variations in the speed and

Fig. 2 P&H TS4100 electromechanical excavator

Fig. 3 P&H TS4100 swing gearbox and motor

vibration spectra, which can be identified by the shape of the speed trace. Among similar groups of these repetitive motions (such as the first and third ranges of steady acceleration and the corresponding spectra in Fig. 4), the response is also similar. This behavior suggests that if a program could automatically identify and extract segments of similar duty, these segments could then be used to isolate and sort data for fault detection. This type of automatic identification of the operating mode could assist the overall monitoring strategy. Section 3.1 proposes and describes the testing of such a strategy.

3.1 Experimental Setup. A National Instruments PXI data acquisition system has been used to collect all video, speed, and vibration data on a P&H TS4100 electromechanical excavator. The vibration acquisition card employed had a resolution of 24 bits at a maximum sampling rate of 102 kS/s, which has been deemed to be more than adequate for the acquisition of signals from the gearbox having a highest known frequency of 2 kHz. Industrial ceramic shear integrated circuit powered accelerometers from IMI Sensors measured vibration in radial directions both horizontally and vertically on both the gearbox and motor bearings. These sensors had a reported sensitivity of 10.02 mV/m s^2 with a claimed error of $\pm 5\%$, over a range of 0.5–6500 Hz. Vibration measurements from these accelerometers have been taken in radial directions both horizontally and vertically on both the gearbox and motor bearings. All measurement channels have been sampled synchronously at 4 kHz.

Figure 5 demonstrates the spatial configuration of the data acquisition system in relation to the main propulsion components in the excavator's housing.

3.2 Methodology. The dataflow for the mode classification program, as specifically applied to the shovel swing machinery, is depicted in the flowchart in Fig. 6, adhering to the general classification approach described in Fig. 7. Sections 3.3–3.7 provide a detailed description of the operation of this program.

3.3 Collection of Data Set for Testing. A labeled data set is necessary to train and test the mode identification classifier. These data are collected over a period of 45 h on an operation of a P&H TS4100 electromechanical excavator operating in normal conditions, using a portable data collection system. The data set started with approximately 300, 1 min records, representing a period of normal fault-free operation. A manual sorting and tagging of the data by mode of operation yielded approximately 160 examples of labeled speed profiles of the shovel swing behavior. The records are further sorted manually into two groups. The first group, which is the loaded swing, represents examples of the shovel swinging away from the bank with a full bucket of oil sand. The second group, which is the unloaded swing, represents examples

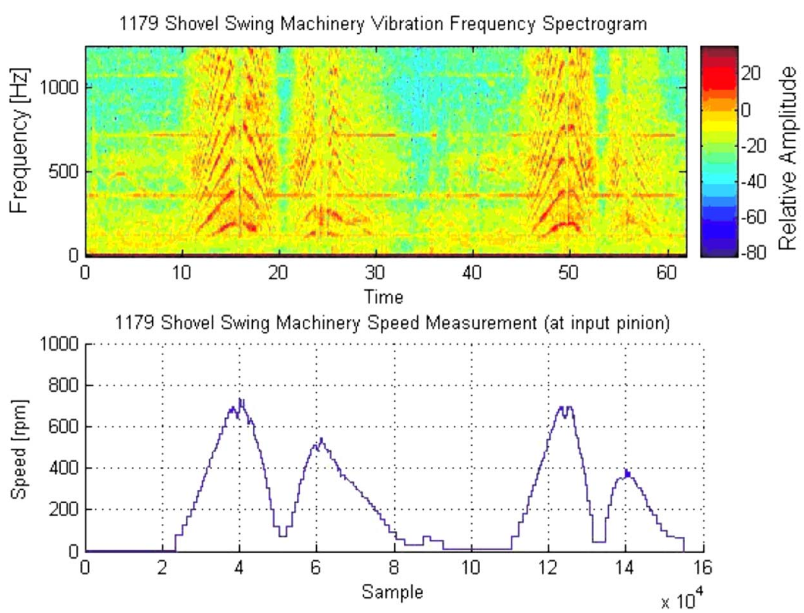


Fig. 4 Spectrogram (top) and corresponding speed plot (bottom) of gearbox

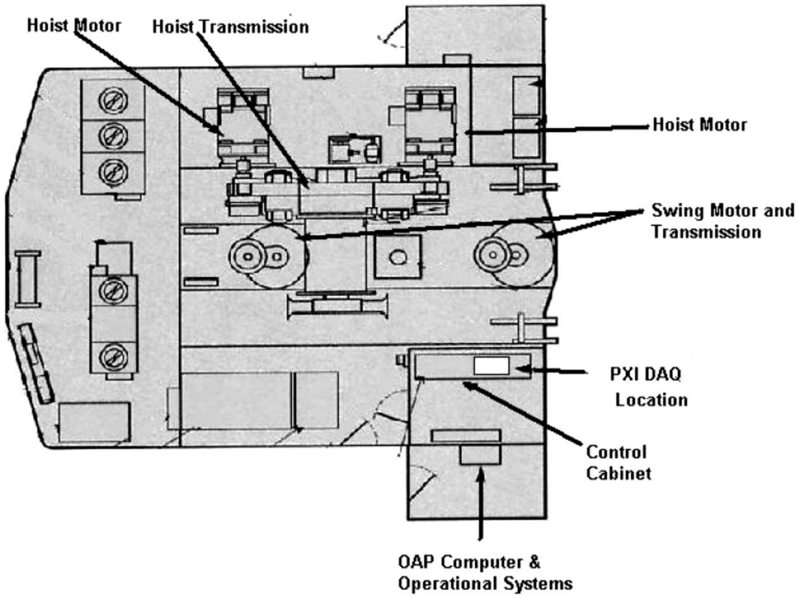


Fig. 5 Plan view of machine enclosure for P&H TS4100 excavator

of the bucket returning to the bank with an empty bucket. It is these two groups of operation that the classification system will endeavor to identify.

It is an inherent assumption of this approach that buckets are either empty or nearly full. The economics of an oil sand operation demands that this be the norm; shovels operating with less than full loads are not practical in an environment where there are a limited number of excavators available to maximize production. While it is entirely possible that some fractional loading may occur, the occurrence of such an event is unlikely, and significant deviations from the fully loaded state will result in the classifier rejecting the mode. Rejection of the mode will not generate erroneous error alarms as the technique employs a moving average of alarm levels—requiring an alarm state to exist for a relatively extended period of time before an alarm is generated.

3.4 Choice of Signals. Vibration and speed are chosen as the parameters on which the mode classification would be based. Using these common diagnostic parameters allows the same basic program structure to be applied directly to other applications (such as hydraulic swing machinery or even the planetary final drives on a haul truck). The common denominator between applications is the vibration signals coupled with an accurate synchronized speed measurement.

3.5 Segmentation of Signal. Segmentation is the first key task of the classification system. Its role is to separate individual discrete segments or events from within a continuous signal. For an online condition monitoring application, this task requires a priori knowledge of the system and can be accomplished with the use of a logical rule base of some sort. In the particular case of the shovel swing machinery, the parameter upon which the segmentation is based is the speed of the gear train. A simple program (described in Fig. 8) is written to segment the data based on the pulse-width measurements of the tachometer pulse signal from the gearbox coupling. The algorithm is illustrated in Fig. 9.

3.6 Extraction of Temporal Features for Pattern Classification. This portion of the program extracts features from the segmented speed profile that reveal information about the operating mode of the shovel. These features will be processed by the classifier to estimate the machine's operating mode.

To generate a smooth speed profile suitable for shape-based feature extraction, the pulse-width measurements needed to be time mapped and smoothed out using a spline approximation fit, constrained to zero slope at each end. The result is a continuous curve approximation fitted to the speed calculations and plotted at even time increments, as shown in Fig. 9. The circles represent the individual speed measurements, while the curve shows the spline fit of the speed profile with zero slope at either ends.

The features chosen for the mode classification are listed in Table 1. These particular features, relating to the shapes of the speed curves, are selected for their computational efficiency and applicability to this specific problem. The speed and vibration features are extracted from the entire speed and vibration segments, respectively. Once measured, the features from the same speed segment are assembled into a single feature vector, which is then passed on to the classification algorithm.

3.7 Numerical Feature Extraction and Classification. The function of the numerical feature extraction and classification algorithm is to determine the class membership of the unlabeled feature vectors. This study examines and compares several approaches to this task. Three commonly used techniques for preprocessing feature extraction are compared: principal component analysis (PCA), independent component analysis (ICA), and exhaustive feature selection (EFS). The algorithms have been implemented using the classification toolbox for MATLAB® [9]. Five supervised linear and nonlinear classification techniques of different types have been compared. These are as follows: least-squares, C 4.5, nearest neighbor, radial basis function network, and support vector machine (using a radial basis function kernel). The various classifiers and feature extraction methods are listed in Fig. 10. Preprocessing of the feature vectors is performed to reduce their dimensionality for the classifiers. This step also made it possible to graphically visualize the data distributions and clustering.

4 Testing and Results

This section describes the results from testing the mode classification system on various combinations of steps for feature vectors, segmentation parameters, preprocessing algorithms, and classification algorithms. The variations of the mode classification system that were tested are shown in Fig. 11. Each trial used a different set of four techniques from the four steps.

4.1 Preprocessing Algorithm. Of the three preprocessing algorithms that were tested, only EFS was able to consistently sepa-

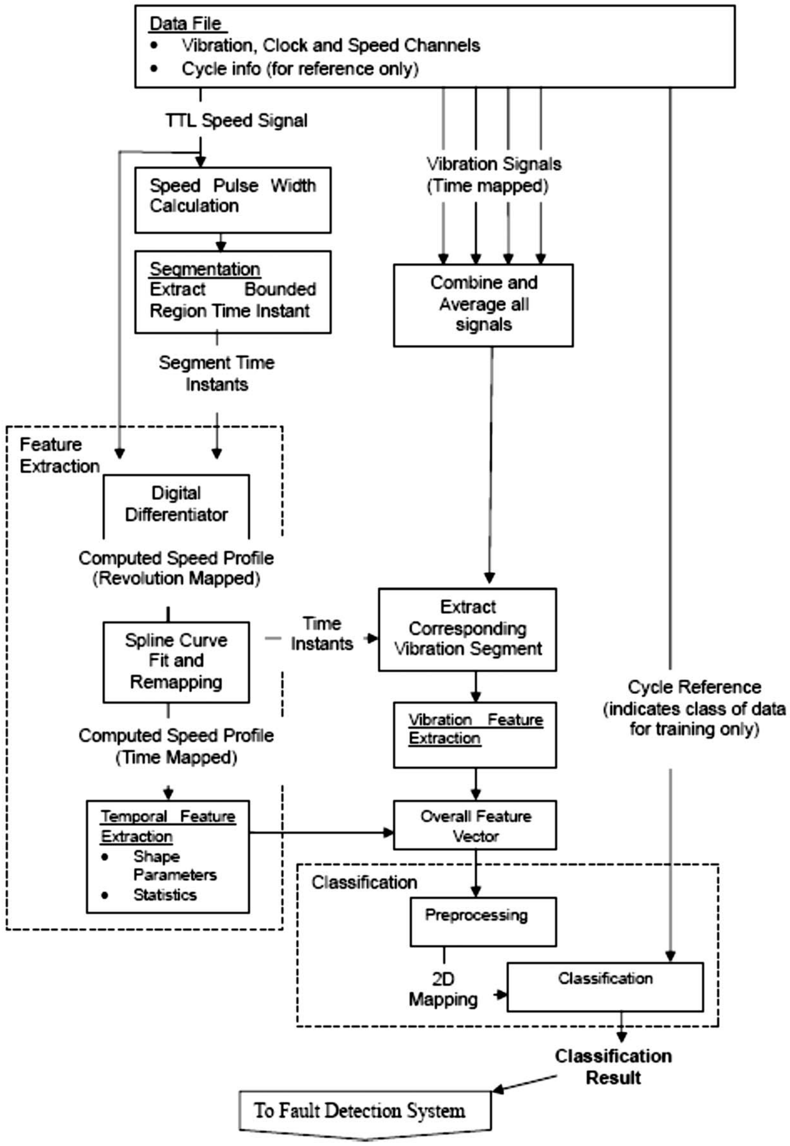


Fig. 6 Mode classification data flow using speed and vibration features

rate the data into distinct regions. In the PCA and ICA distributions, there is a significant amount of overlap between the two classes. As would be expected, the poor performance of the PCA and ICA algorithms in separating the two data groups influenced the error rates of the classifiers as well.

Principal component analysis seeks to project a high-dimensional data set onto a lower one in an optimal fashion. The technique first finds some favorable point in the high-dimensional space that will serve as the "anchoring point" for a lower-dimensional hyperplane whose orientation will minimize the dimensionality-reduction representation error. The vectors defining the orientation of this plane define the "principal components" onto which the data are projected to transfer them into the lower-dimensional space. This process may be viewed in an alternative fashion, as seeking out a direction, which provides maximum correlation between the features. The inability of PCA to develop a completely separable lower-dimensional representation may then be attributed to a lack of any best orientation of the lower-dimensional hyperplane in relation to its anchoring point (usually taken as the data's mean vector). This may in turn be attributed to the choice of feature parameters that lack such a correlation. In selecting these features, domain knowledge was used to select features that best represent the changes in machinery state; for the transducers employed, these are clearly speed and some aspect of the vibratory response of the machinery. There may exist some, as

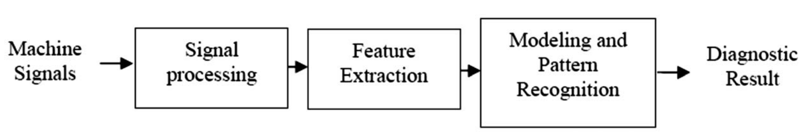


Fig. 7 General pattern recognition system for condition monitoring

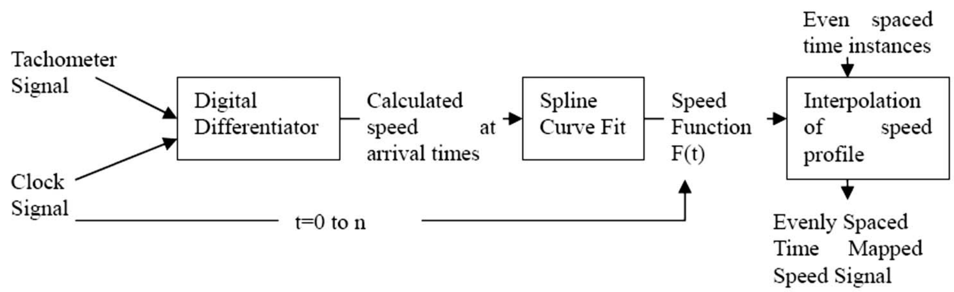


Fig. 8 Calculation of speed profile

yet unknown, optimal subset of these two factors, which will produce a perfect PCA representation, but in real-world applications, one must always expect some degree of overlap.

While PCA seeks a direction, which best represents the data in a lower-dimensional space, independent component analysis seeks a representative hyperplane, whose components are maximally independent from one another. ICA is a logical improvement over PCA, since in pattern recognition we attempt to define boundaries that best separate data; if the space onto which they are projected is already fairly dissimilar (from the ICA process), then we may produce a representation that is better suited for classification. The poor performance of the ICA algorithm likely stems from the similar shortcomings of the PCA. An ideal feature set may exist, which once reduced produces perfectly separable data, but achieving such a feature set with real-world data may not be practical.

With these two predominant approaches explored and discarded due to poor performance, a technique that will optimize lower-dimensional representation with some properly motivated algorithm seemed out of reach. Instead, exhaustive feature selection is used to seek combinations of features, which will produce minimal classification error. Exploring combinations of features and using their resultant classification errors as the selection criteria in a branch-and-bound approach are appropriate for such contexts where rationally motivated approaches (such PCA and ICA) fail. It is also reasonably well suited to automation without user intervention.

When comparing across all combinations of the other parameters, EFS yielded a more accurate classification result than either PCA or ICA. PCA yielded the highest error (approaching a random result of 50% in several cases). The results of how the classifiers performed, according to preprocessing strategy, are summarized in Fig. 11. Clearly, the poor separation of the two data groups, as shown with PCA, makes a difficult classification task.

4.2 Feature Choice. The results were re-examined after discarding the combinations of classifiers and preprocessing schemes that performed poorly overall. Figure 12 is a comparison of the classification performance averaged across all of the classifiers using EFS. These results suggest that the acceleration mode segment was slightly easier to classify than the deceleration mode segment. These results would agree with the subjective observation that the acceleration portion of the shovel's swing is smoother and more repetitive than that of its deceleration. Figure 12 also reveals that the addition of the vibration features to the overall feature vector increased the accuracy of the classifier. However, this advantage was less pronounced for the acceleration features than it was for the deceleration features. This difference in variability can be explained by the nature of these activities. Acceleration at the start of swinging is a standard commanded trajectory no matter what the digging circumstances, but deceleration requires judgment about trajectory planning to stop over the target location (the truck into which the ore is dumped).

For certain preprocessing algorithms and feature sets, after going through the preprocessing stage and being reduced in dimension down to two, there was a marked difference between the clustering of the data from the swinging-full and swinging-empty

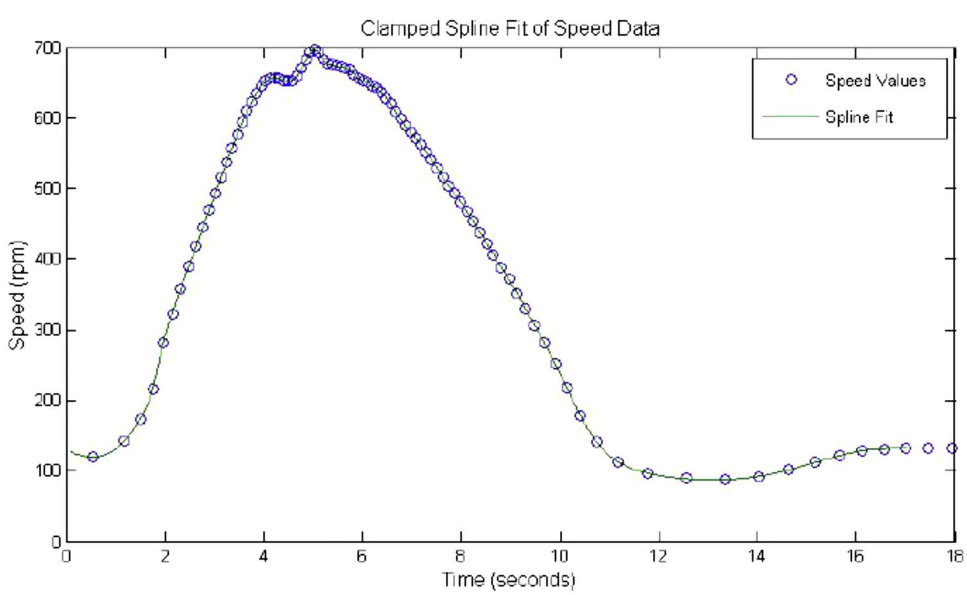


Fig. 9 Spline fit of speed profile

Table 1 Features extracted for mode classification

Feature	Description
S1	Numerical integral of speed curve (related to amplitude of swing derived from integral speed sample)
S2	Average of numerical derivative of speed profile
S3	Maximum of numerical derivative
S4	Minimum of numerical derivative
S5	Absolute value of range of numerical derivative (S4–S3)
S6	Mean speed during segment
S7	Maximum speed during segment
S8	Median speed during segment
S9	Average speed during segment
V1	Statistical skewness of vibration time signal. An indication of symmetry and corresponding to the third-order moment about the mean.
V2	Kurtosis of vibration time signal. An indication of the peakedness of the signal corresponding to the fourth-order moment about the mean.
V3	Root mean square of vibration time signal
V4	Peak to peak of vibration time signal

states. The swing-empty class of data is closely clustered. In other cases, the two data groups clustered in about the same density. Possible explanations include the following:

(1) Variations in the loading of the swing machinery between different examples of the "full" data class. The amount and density of the ore in the bucket has an effect on loading. This hypothesis is further supported by the fact that in comparison to the "swing full" class, the swing-empty class would have negligible fill variation. In other words, empty is empty but full has variability in the classification.
(2) Another possible explanation could be due to the nature of the operator's control task of swinging full (and dumping) and then returning to the bank for more digging. Swinging out to the truck with approximately 100 tons of ore and having to accurately place the load would appear to be a more critical task. In contrast, swinging back to the bank with an empty bucket would appear to be more repeatable. Whatever the underlying reason is for the differences in the two data sets, these results suggest that for an operational system, the swing-empty cycle would provide more consistent results. Using this single mode would allow the use of a one-class classification scheme for mode identification.

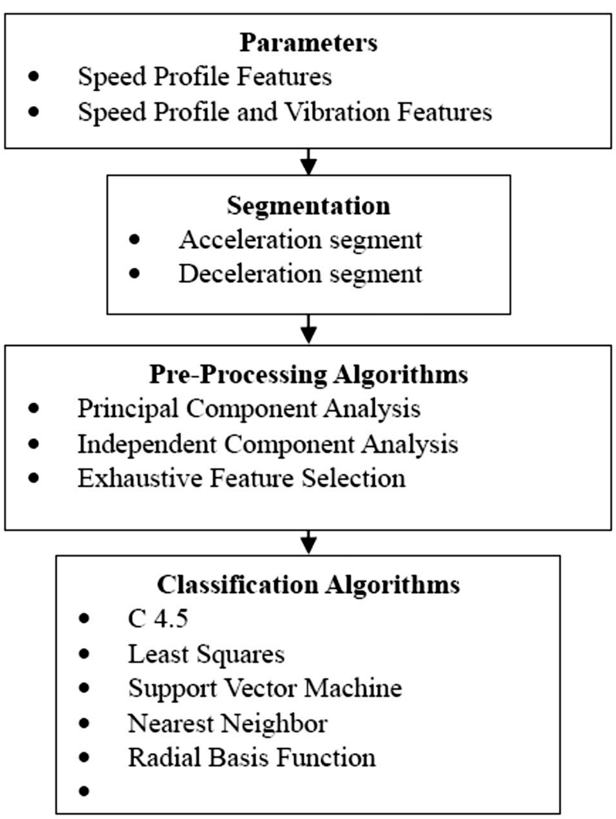


Fig. 10 Combinations of system configurations used for testing

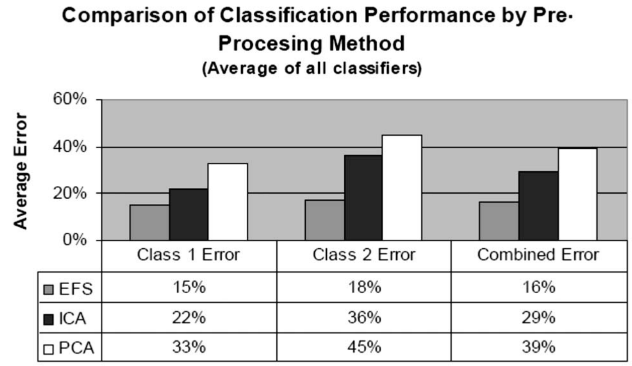


Fig. 11 Comparison of preprocessing methods (averaged across all features and all classifiers)

4.3 Classification Algorithm. The ability of the mode classification program to perform its task was quantified using the classification error. This error is defined as the proportion of the data that is assigned to the wrong category by the classifier. The test error rate is a measure of the trained classifier's ability to classify new (previously unseen) data, while the training set error shows how well the classifier fits the training data.

The abundance of example data allowed for different data to be used for training and testing. In these tests, the 80% holdback method was used, where 20% of the data were used for training of the classifier and 80% of the data were used for testing. The data labeled as class 1 and class 2 represent the empty and full swing cycles, respectively. The classifier performance for all of the combinations of the chosen segments, features, preprocessing, and classification algorithms is summarized in Tables 2 and 3. The best performing classifier/preprocessor combination for each of

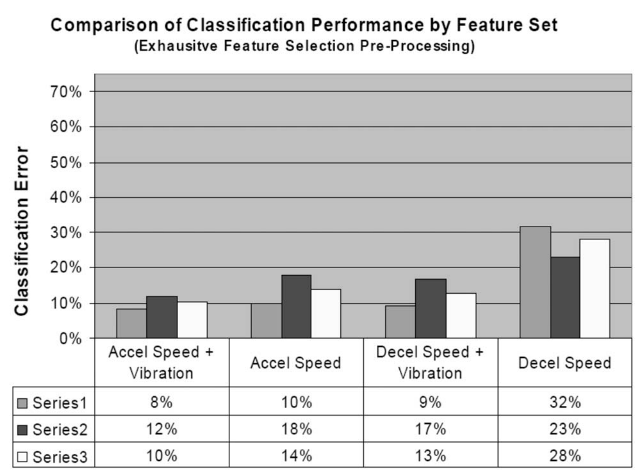


Fig. 12 Feature set comparison (averaged across all features and all classifiers)

Table 2 Classification results for acceleration segments

Feature set	Classifier	Test set errors			Training set errors		
		Class 1	Class 2	Average	Class 1	Class 2	Average
Speed and vibration	C 4.5	0%	16%	8%	13%	14%	13%
	LS	10%	9%	10%	13%	0%	7%
	SVM	16%	3%	10%	0%	0%	0%
	N.N.	**9%**	**3%**	**6%**	**0%**	**0%**	**0%**
	RBF	6%	29%	17%	0%	0%	0%
Speed only	C 4.5	18%	3%	11%	0%	11%	7%
	LS	6%	20%	13%	0%	0%	0%
	SVM	**10%**	**13%**	**11%**	**0%**	**0%**	**0%**
	N.N.	9%	21%	14%	0%	0%	0%
	RBF	7%	32%	21%	0%	20%	7%

the four feature sets is emphasized with bold font.

Graphical representations of typical two-dimensional decision boundaries for the different classifiers are shown in Fig. 13. The darker region denotes the area designated by the empty swing area. For well-separated data (as was the case with EFS preprocessing), the linear decision boundary did a reasonable job at separating the two classes.

The output of the preprocessing stage would ideally produce completely separable data classes; in practice, this is not possible and a variety of different classification techniques have arisen in order to address the resultant varying needs. It is an accepted principle in pattern recognition that there is no best classifier for all applications. Acknowledging this, the authors set out to find the most suitable classifiers that could be employed in an automated fashion (i.e., those requiring the least amount of configuration). Some techniques such as complex neural networks require a great deal of experience in deciding the configuration of classification parameters (e.g., number of layers and number of nodes per layer) and as such are poorly suited to automation. Other techniques, such as least-squares discriminants, support vectors, nearest-neighbor classifiers, and the C 4.5, have few magic parameters and are eminently qualified for automated modal classification, as shown in Fig. 13.

The data dispersion in Fig. 13 does not appear to separate linearly, and so it seems that it might not be a suitable application for least-squares discriminants. However, classification with support vector machines (SVM) produces approximately the same error as the least-squares approach. With the goal of automating classification as much as possible, this comparable performance suggests that least-squares discriminants are suitable to automated operating mode classification with only a minor comparative difference in performance for this application.

The SVM classifier is a very flexible classifier with a minimal number of "magic" parameters for configuration and automation. The SVM is designed to separate data classes optimally in a kernel space in order to separate them in their linear input space. This optimization can sometimes leave patterns on the wrong side of a classification boundary as the goal is to minimize overall error, but attempting to ensure that each training pattern is classified correctly may lead to over fitting. The kernel function and kernel parameter employed in the SVM has a profound impact on this consideration. If a Gaussian kernel function is chosen, a small kernel parameter will lead to overfitting, while a large kernel parameter will lead to overgeneralization. The error on the training set induced by the optimization technique of the SVM as well as the choice of kernel parameter is shown in Fig. 13. Some training objects are misclassified, but the result is an error on par with some of the better classifiers employed. In the context of automated mode classification, the SVM whose results are shown in Fig. 13 produces an agreeable balance of classification error and classifier complexity appropriate for the task.

The radial basis function (RBF) network's classification results are presented in Fig. 14, as a basis for comparison to the other techniques that are suited for automation. Bennett and Campbell [10] notes that neural networks are riddled with classification issues, such as potential for minimization techniques encountering local minima, many parameters to configure, a lack of stability, reproducibility, and specific algorithm independence. It is no surprise that the RBF network, whose results are presented in Figs. 13 and 14, performed poorly. The finicky nature of the classification makes it difficult to pinpoint the source of error; neural networks are generally expected to perform as well as SVMs only when specialized optimization techniques are employed (see Ref. [11]).

A comparison of the classifiers on the acceleration data is shown in Fig. 14 except for the RBF network on class 2 data; all of the classifiers performed reasonably well on the data that has been preprocessed using EFS. The nearest-neighbor classifier per-

Table 3 Classification results for deceleration segments

Feature set	Classifier	Test set errors			Training set errors		
		Class 1	Class 2	Average	Class 1	Class 2	Average
Speed and vibration	C 4.5	**13%**	**7%**	**10%**	**14%**	**0%**	**7%**
	LS	9%	19%	14%	0%	13%	7%
	SVM	10%	12%	11%	0%	0%	0%
	N.N.	3%	20%	11%	17%	11%	13%
	RBF	12%	27%	19%	0%	22%	13%
Speed only	C 4.5	39%	7%	24%	17%	0%	7%
	LS	**32%**	**17%**	**25%**	**40%**	**0%**	**13%**
	SVM	42%	13%	29%	0%	0%	0%
	N.N.	42%	7%	25%	50%	11%	27%
	RBF	3%	72%	38%	0%	57%	27%

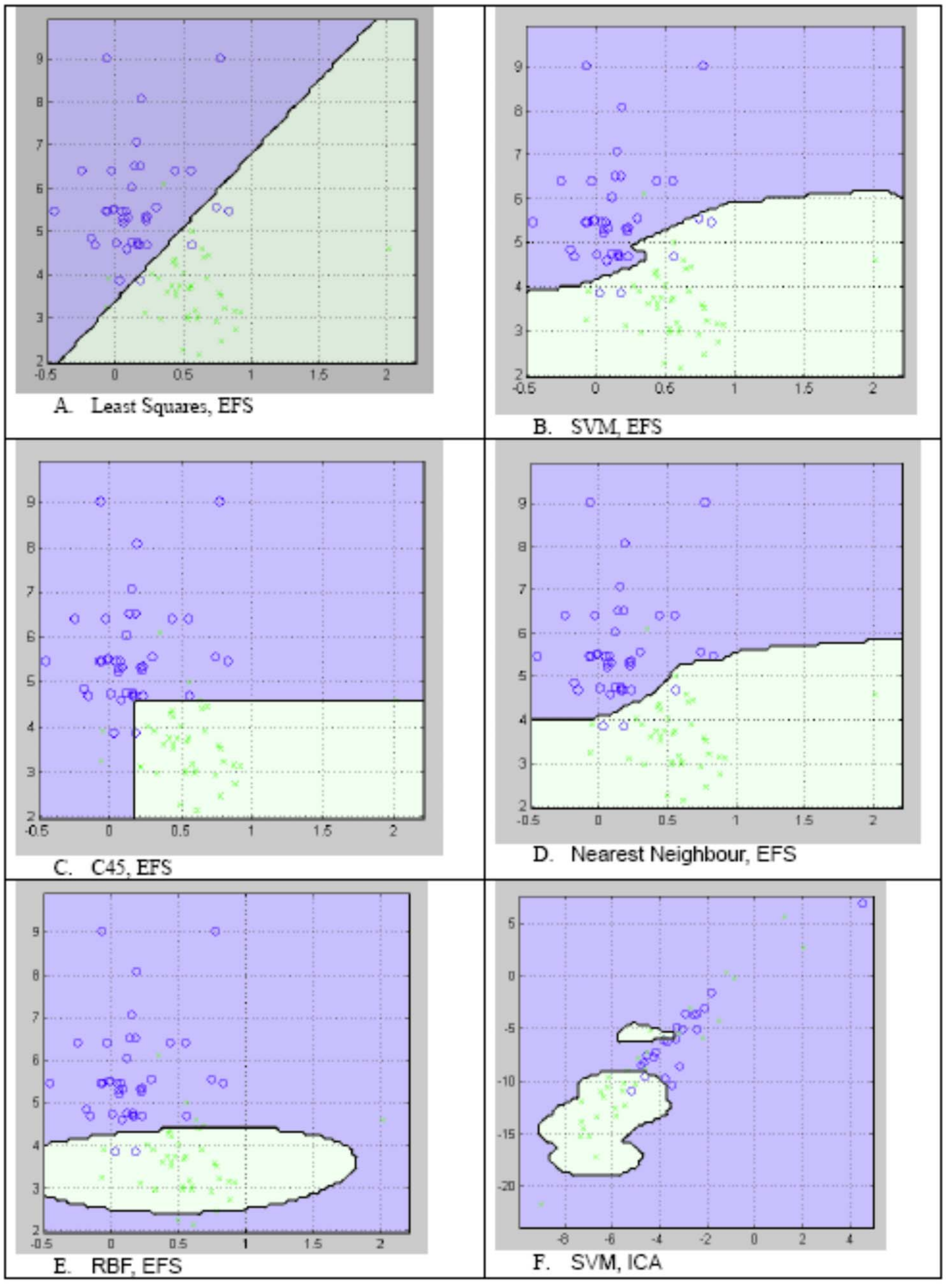
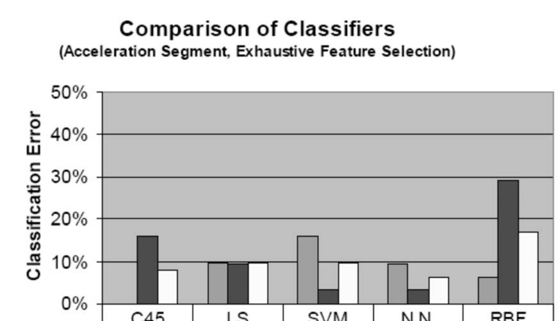


Fig. 13 Classifier decision boundaries, acceleration speed features (0—empty; x—full)

formed the best overall on the acceleration data (Fig. 14), but if class 1 acceleration data were only used, then the C 4.5 algorithm will produce an error-free classification result.

5 Conclusions

These experiments demonstrated that, for the particular case under observation, the proposed method of classifying operating mode using temporal features is a viable one. The experiments also revealed that the selection of classification technique and feature extraction method play an important role in the accuracy and reliability of the classification. The best overall result working on both classes of data was found to be the nearest-neighbor classifier with exhaustive feature selection, having a combined error of 6%. This error rate was as low as 3% if a single class (class 1) of data was chosen. A relatively conservative training and testing approach was used (by holding back 80% of the data for testing). If



	C45	LS	SVM	N.N.	RBF
Class 1 Error	0%	10%	16%	9%	6%
Class 2 Error	16%	9%	3%	3%	29%
Combined Error	8%	10%	10%	6%	17%

Fig. 14 Comparison of classifiers—acceleration segment

more data were used for training, an even lower error rate would be anticipated. Also, none of these models has been fine-tuned, which would have further reduced the error rates. From these results, it is concluded here that, as a part of the overall system, the proposed approach is feasible for classifying the mode for the type of machinery in question using only temporal features from the speed profile.

In future implementations, it is recommended that a similar assortment of alternate techniques will be trained and tested for a particular problem, and the best combination should be chosen from the alternatives. Another step in this research will involve conducting experiments to investigate the influence and benefit of using mode information to augment a fault detection system.

Acknowledgment

The authors would like to acknowledge the financial assistance and technical cooperation of Syncrude Canada Ltd. and the Natural Sciences and Engineering Research Council of Canada (Collaborative Research and Development Grant, CRD 267639 - Development and Integration of Maintenance Decision Support Technologies) for supporting this research.

References

[1] Hayton, P., Scholkopf, B., Tarassenko, L., and Anuzis, P., 2000, "Support Vector Novelty Detection Applied to Jet Engine Vibration Spectra," NIPS 2000.

[2] Baydar, N., and Ball, A., 2000, "Detection of Gear Deterioration Under Varying Load Conditions by Using the Instantaneous Power Spectrum," Mech. Syst. Signal Process., **14**(6), pp. 907–921.

[3] Chen, P., Taniguchi, M., Toyota, T., and He, Z., 2005, "Fault Diagnosis Method for Machinery in Unsteady Operating Condition by Instantaneous Power Spectrum and Genetic Programming," Mech. Syst. Signal Process., **19**(1), pp. 175–194.

[4] Zhan, Y., Makis, V., and Jardine, A. K. S., 2006, "Adaptive State Detection of Gearboxes Under Varying Load Conditions Based on Parametric Modeling," Mech. Syst. Signal Process., **20**(1), pp. 188–221.

[5] Worden, K., Sohn, H., and Farrar, C. R., 2002, "Novelty Detection in a Changing Environment," J. Sound Vib., **258**(4), pp. 741–761.

[6] Sohn, H., Wordern, K., and Farrar, C., 2001, "Novelty Detection Under Changing Environmental Conditions," SPIE's Eight Annual International Symposium on Smart Structures and Materials.

[7] Timusk, M. A., Mechefske, C. K., and Lipsett, M. G., 2005, "A Pragmatic Framework for On-Line Neural Network Based Detection of Machinery," International Journal of Condition Monitoring and Diagnostic Engineering Management (COMADEM), **8**(1), pp. 35–41.

[8] 2005, SKF Corporation website, www.SKF.com.

[9] Stork, D. G., and Yom-Tov, E., 2004, *Computer Manual in MATLAB to Accompany Pattern Classification*, 2nd ed., Wiley, New York.

[10] Bennett, K. P., and Campbell, C., 2000, "Support Vector Machines: Hype or Hallelujah?," SIGKDD Explorations, **2**(2), pp. 1–13.

[11] Samanta, B., 2004, "Gear Fault Detection Using Artificial Neural Networks and Support Vector Machines With Genetic Algorithms," Mech. Syst. Signal Process., **18**, pp. 625–665.