CHAPTER 5

Results

The results from the cutting experiments are shown in this chapter. The techniques for signal processing shown in previous chapters are implemented and classification is done using HMM techniques. This is compared with results from a Bayesian classifier.

5.1 Wear progression

One of the problems with measuring in machining environments is the adverse conditions. To shield the delicate strain gauges an epoxy covering was applied over the gauges. This may sometimes have the effect that the strain gauge comes lose from the tool holder. This happens when the epoxy covering constrains the strain gauge during large strains and causes a complete tear of the strain gauge glue. The strain gauge is then completely lose from the tool holder. In such a case the measuring device becomes completely useless and has to be replaced.

Such a "release" of the strain gauge happened during this experiment. The photos from the microscope suggest that the tool has worn from $0mm$ to $o.1mm$ at its nose. This is a third of the usually allowable $0.3mm$ for flank wear and represents about a third of its useful life. It was decided to use the data from an incomplete tool life anyway. The rationale for this is, that if very accurate classification can be achieved at this stage, then surely better classification will be achieved with full tool life data.

If a full tool life is available then, using the same techniques, more wear levels can be appointed. This will ultimately point the system in the direction of a continuous wear estimator.

For the rest of the classification procedures, this wear level will be referred to as the "worn" condition.

5.2 Signal processing

5.2.1 The raw signal

Software for the recording of the signals were written in a such a manner that a software trigger can be set to initialise the recording process. This was set to start recording 1 second before cutting starts. Figure 5.1 shows a typical cutting signal in the feed direction. The little 1 second buffer in the beginning of signal can be seen. This part of the signal has its use to categorise noise of the system when no cutting is taking place. A noise filter can be implemented from information of this noise.

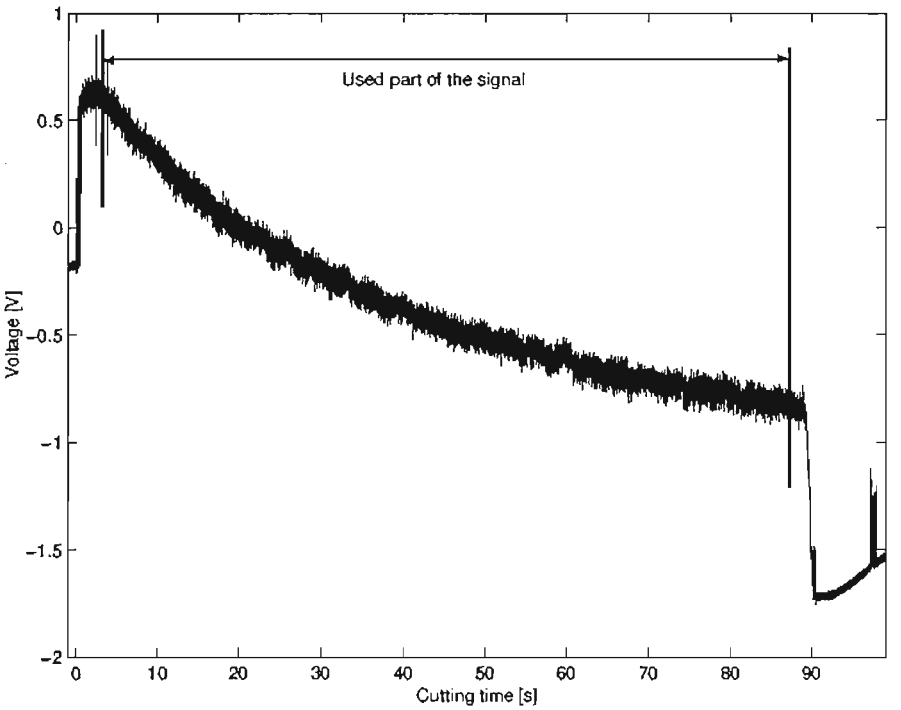

Figure 5.1: A typical cutting signal from the feed direction.

Since the cut is always made to be approximately the same length, a collar is formed on the workpiece. When the insert exits the workpiece, it "rubs" against the workpiece. The erratic nature of the last part of the signal is therefore caused by the exit procedure of the insert from the workpiece.

Another interesting observation from figure 5.1, is the effect of temperature at the tool tip on the signal. This effect can be seen as by the phenomenon that looks like an exponential decay on the signal. If recording of the signal was continued till long after the cut, the signal returned to a position very close to the original zero. This seemingly suggests a first order response which is indicative of temperature effects. This temperature response may be useful for the monitoring of tool wear in further studies,

but it is not nearly consistent enough. This "cutting temperature" is a strong function of the depth of the cut, and although it has been argued that inconsistencies in the data may prove the system to be more robust, the cutting temperature is considered to be too sensitive for practical use.

5.2.2 Segmentation and preparation

The signal shown in figure 5.1 is as such not yet very useful and still needs to be processed into a usable form. The first step was the removal of the temperature effects so that all digital drift effects are removed. This is done by removing a linear trend from the data so that the start and end of the signal are at the same voltage.

No use will be made of the transients at the beginning and end of the signal and they will also be removed. The transients may also contain valuable information but there are only two transients and their length is less than a thirtieth of the total signal length. The focus and emphasis will be on the continuous part of the cutting signal which is easier to monitor and segment.

Since the mean of the signal is very dependant on the depth of the cut it will also not be used for the processing since this features was very much influenced by the operators own expertise. This makes the segmentation and the removal of the temperature effects and the mean very easy. All signals will be segmented as shown into the useful parts as shown in figure 5.1. After this the signal is detrended thus removing the dominant linear trend. `Detrend` is a standard MATLAB function that is often used for processing of data for FFT analysis. This was done piece wise to ensure that the signal had a mean of zero. The remaining signal looks like figure 5.2. Figure 5.3 shows a magnified region of figure 5.2.

To show that these signals still carry information and are not just random noise signals, figure 5.4 is provided. This shows a scatter plot of two signals removed by some time. The plot has an oval shape which means that the variance of one of the signals has increased. This is also shown on the histograms plotted on the figure. These histograms have the tops of the bins connected to form a curve. They are also normalised in order to fit into the figure. These histogram have therefore no correlation with the figure axes. The two figures were normalised with the same factors. The figure was aimed to prove that there was still useful information captured in the signals.

To facilitate on-line monitoring the signal is, after detrending, segmented further again into smaller "snippets." It is on these snippets that feature extraction will be done. Each sample in the observation sequences is composed of the features of these snippets. Because frequency domain features are extracted, the snippets needs a certain length in order to contain a usable frequency resolution when the FFT is calculated. A length of $2^{11} = 2048$ was chosen, rather arbitrarily so that, with a sampling rate of $f_s = 20kHz$, a

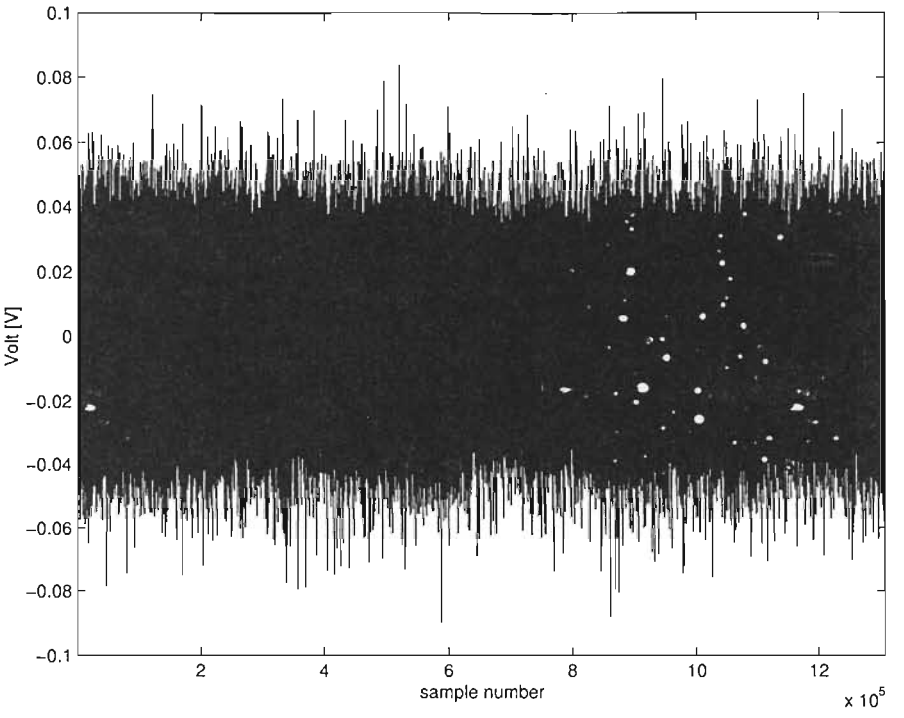

Figure 5.2: The final signal after segmentation and detrending.

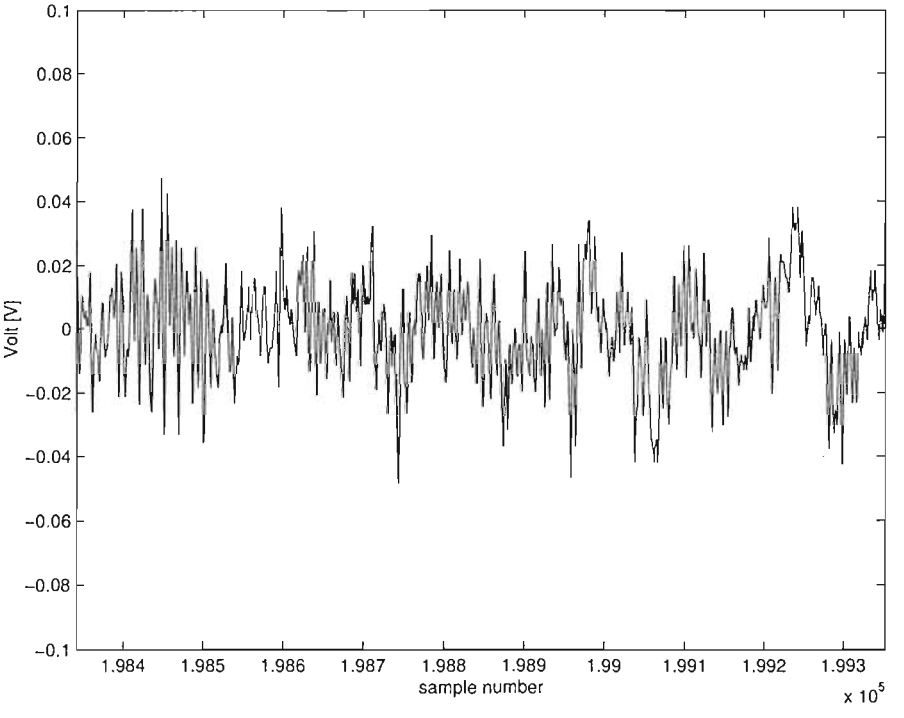

Figure 5.3: A magnified region of figure 5.2

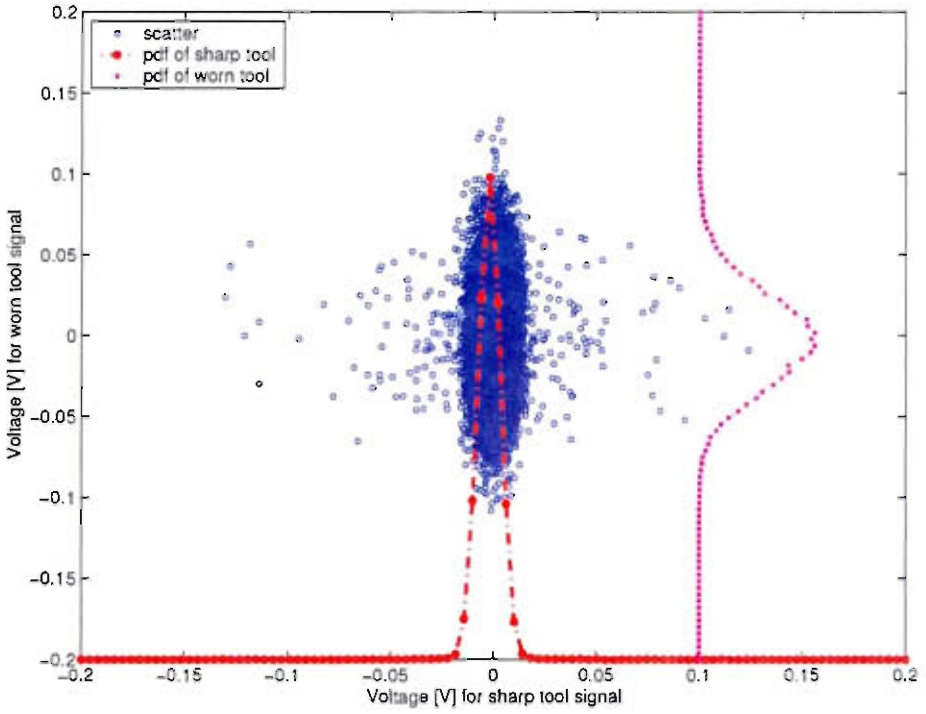

Figure 5.4: A scatter plot of two signal to show the increase in variance.

frequency resolution of $f = 9.76Hz$ can be achieved.

5.2.3 Critique on signal processing results and signal quality

It has been mentioned that the experimental conditions were not kept exactly constant. This was because of the nature of the machining process and the ability of the machine operator. Another interesting factor which has not been mentioned up to now, is that of the workpiece material. When the feature space is viewed, there are certain discontinuities in the signal. After the discontinuities the signal seems to follow a stationary trend until the next discontinuity. These regions each signify a new workpiece. The setup of the workpiece together with differences in chemical composition may be the cause of this.

Figure 5.5 show a pure noise signal produced by the machine. A normalised histogram is shown on top of the noise signal. Clearly the noise of a Gaussian nature and slightly skewed to the lower values. The signal to noise ratio can be calculated from this signal and the one in figure 5.1. Using the equation:

$$S = 20 \log_{10} \frac{RMS_{signal}}{RMS_{noise}} \tag{5.1}$$

In equation 5.1, S is the signal-to-noise ratio , in decibels, of the root mean squares of the noise and the signal. RMS_{noise} was calculated to be 0.005 and $RMS_{signal} = 0.0168$. S can then be calculated to be $S = 10.5dB$. This can be regarded a rather noisy signal if it

is considered that FM radio transmissions may have signal-to-noise ratios of $S = 50dB$. This all implies that for future work a noise filter might be applied to great advantage on this system.

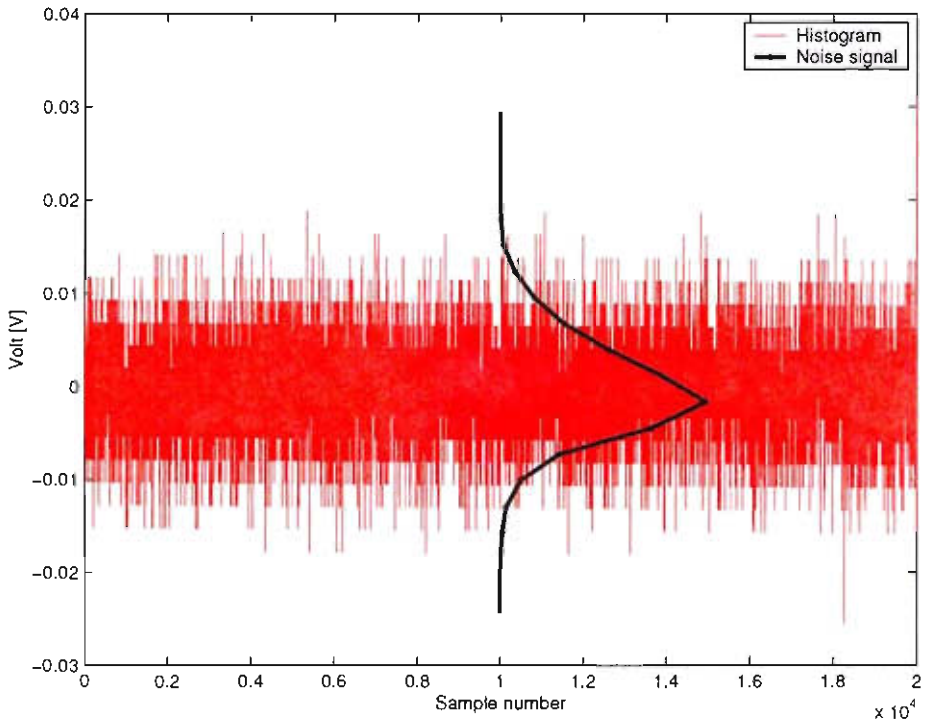

Figure 5.5: A noise signal from the system. Superimposed on the signal is a normalised histogram.

5.3 Feature selection and dimensional reduction

Features from the time domain

Figures 5.6 to 5.7 shows the features that were calculated from the snippets of the de-trended signals. A total of twelve features were extracted, six from the time domain and six from the frequency domain. The sample number is indicated on the x-axis of the feature-figures. Each of these samples represents a time interval for which the feature was calculated.

Figure 5.8 shows the PSDs of the cutting signals of one tool. The frequency axis is displayed up to the cut-off frequency of the anti-aliasing filter. This figure shows some peaks in the range below $300Hz$. These peaks are magnified in figure 5.9. This plot also shows a dotted line which is a sum of all the PSDs. (The sum is divided by a factor 100 in order to make it visible on this plot) This is helpful for finding regions where there is more energy present. Magnifying one of these regions shows the increase of one of the regions. The legend explains how the colour of the line is connected to the time "into" the

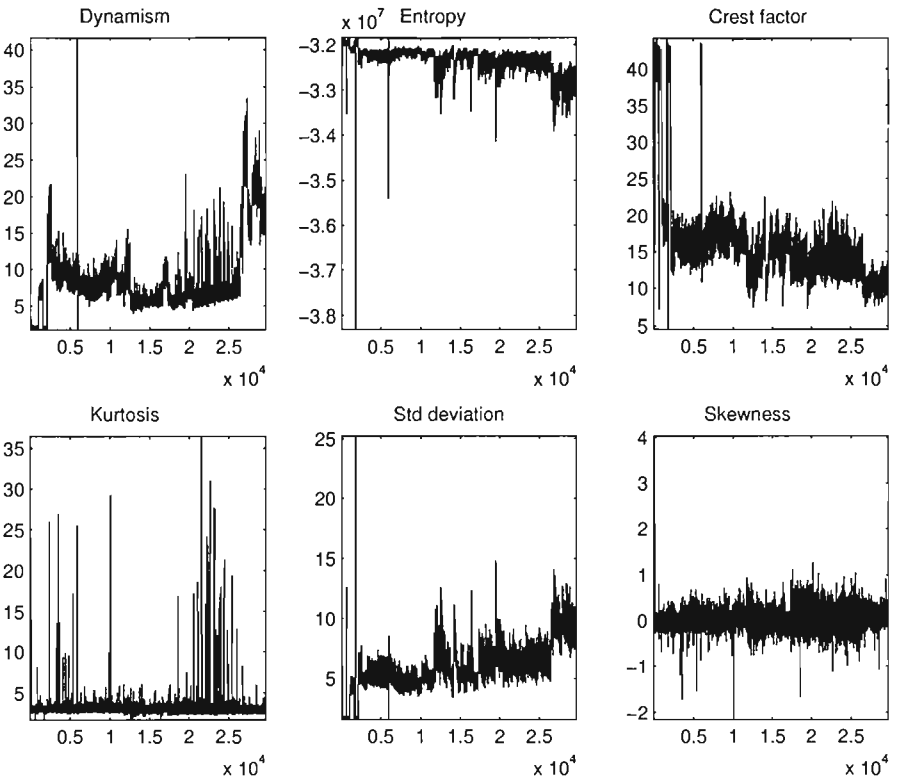

Figure 5.6: The time domain features extracted from the processed signals.

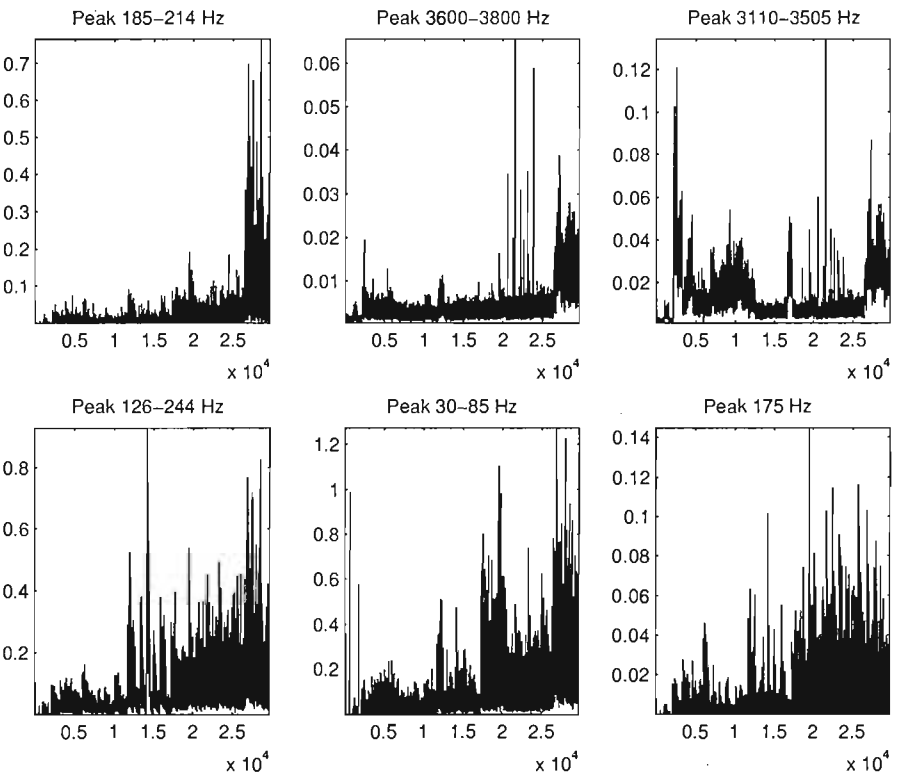

Figure 5.7: The frequency domain features extracted from the processed signals.

tool life. Peaks that may be used will then start out as a small dark hump and gradually "transform" via grey into a light grey peak. Peaks that shrink via this same process are also useful. Figure 5.10 represents these transformations of the peaks in figure 5.9.

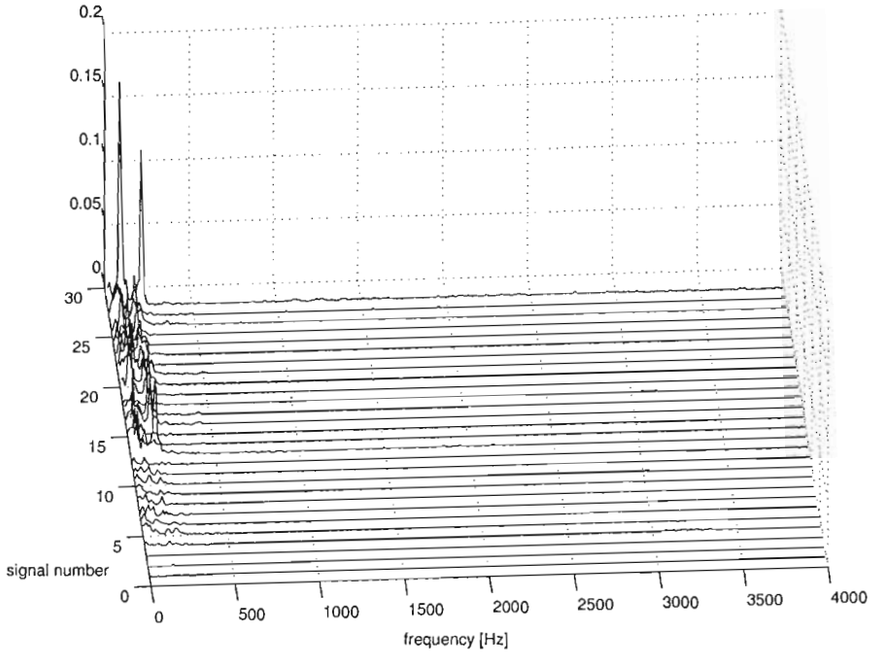

Figure 5.8: The PSDs of the cutting signals during the life of a tool.

5.3.1 The selection process

The selection process is shown in figure 5.11. Features were selected according to their relation to the theoretical tool wear function calculated by the correlation coefficient. The ideal trend function is then taken as a straight line with a slope of 40°. This slope was chosen arbitrarily and any positive figure may be used for this. This slope is chosen to be more steep than any of the slopes from the other features. This is then helpful to select the features that have most consistent correlation with the theoretical tool wear function. The effectiveness of the feature selection process is subsequently dependant on the assumption that the tool wear can be approximated by a straight line. This technique was proposed by Scheffer and Heyns (2001).

The sorted correlation coefficients are shown in table 5.1. Entropy and Crest factor were chosen from this list because of their obvious inverse relationship with the theoretical tool wear. It seems that only the skewness, the kurtosis and the $3110 - 3505Hz$ peak are unusable in this application because their correlation coefficients are a whole order less than that of the rest of the features. Selecting both negative and positive coefficients has the advantage of ensuring that the selected features contain minimal mutual information.

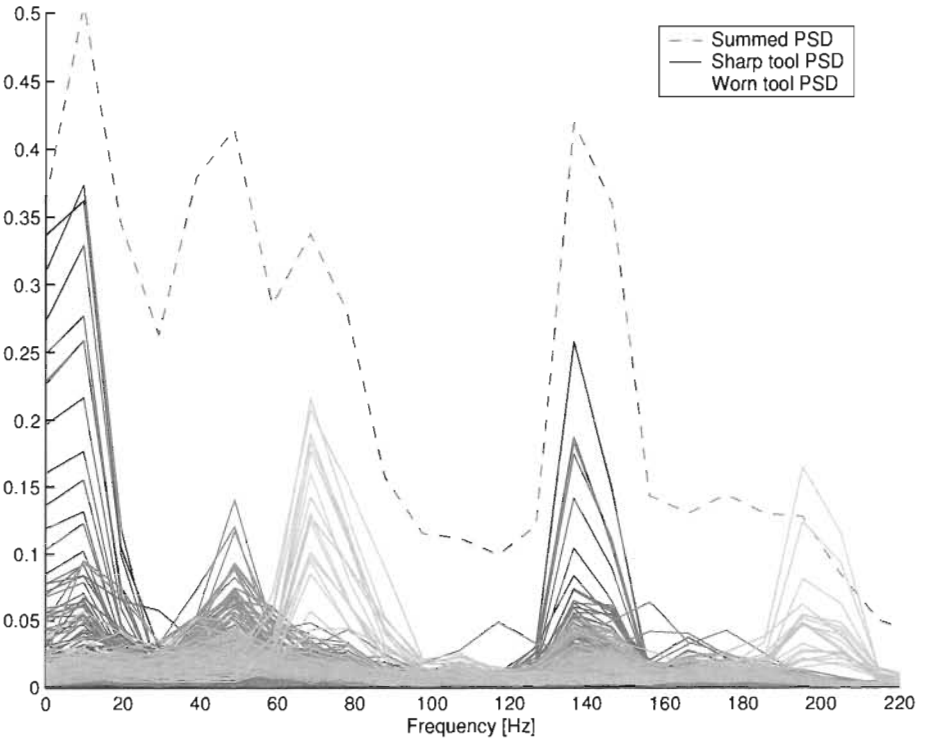

Figure 5.9: The magnified region and the summed PSDs.

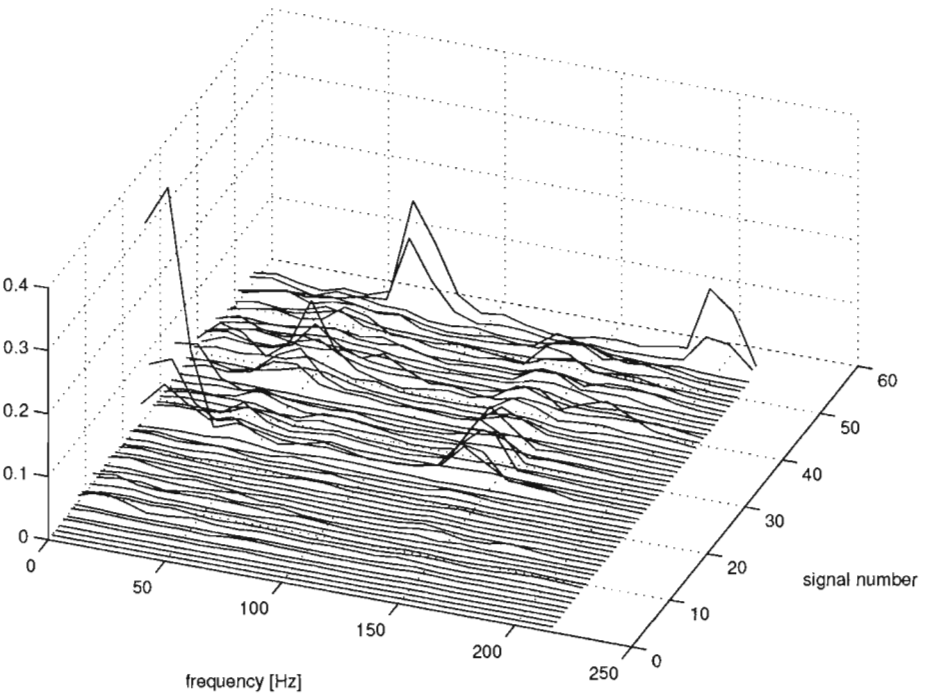

Figure 5.10: Another view of the progression of the PSD peaks.

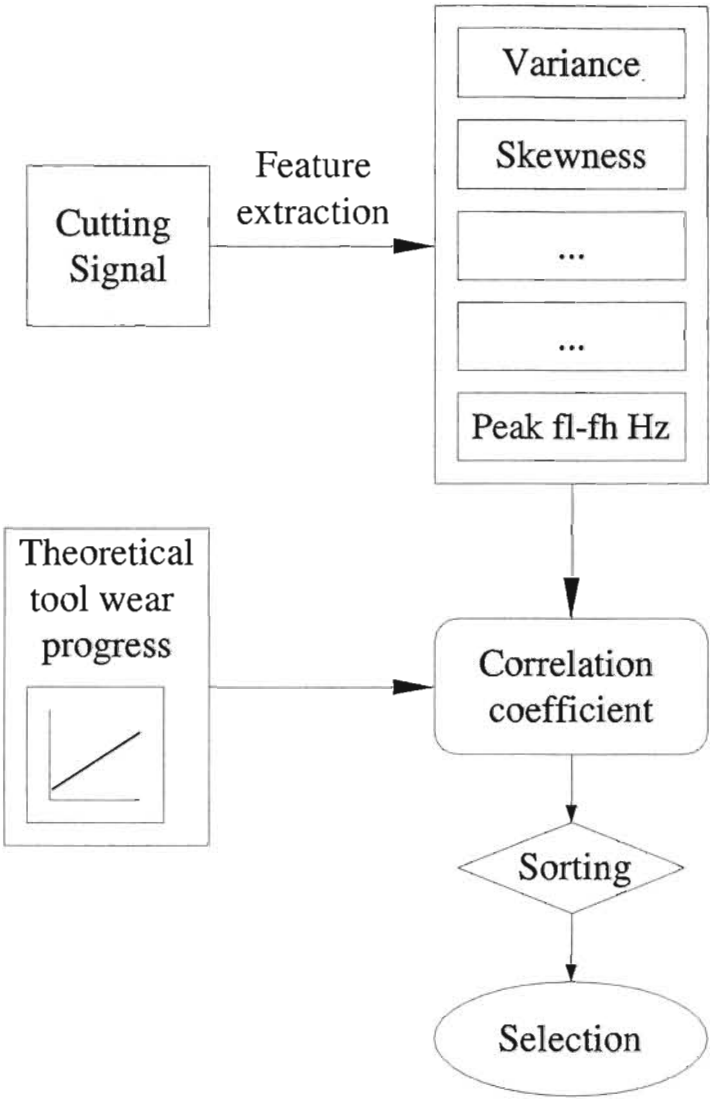

Figure 5.11: The selection of the features using the correlation coefficient.

Table 5.1: The sorted correlation coefficients.

Feature	Correlation Coefficient
Entropy	-0.651
Crest factor	-0.596
Kurtosis	-0.057
Peak 3110-3505 Hz	0.037
Skewness	0.085
Peak 175 Hz	0.379
Dynamism	0.405
Peak 185-214 Hz	0.445
Peak 126-244 Hz	0.516
Peak 3600-3800 Hz	0.540
Peak 30-85 Hz	0.601
Std deviation	0.704

All the other features, except for the above named three were used for the classification process. The final selection of features are listed in tabel 5.2.

Table 5.2: The selected features

Final features
Entropy
Crest factor
Peak 175 Hz
Dynamism
Peak 185-214 Hz
Peak 126-244 Hz
Peak 3600-3800 Hz
Peak 30-85 Hz
Std deviation

Preparation for the feature reduction

Before a feature reduction can be done a whitening transform is done on the data. After the whitening transform all the features in the data have a mean of 0 and a variance of 1. This transform is similar to the normalisation techniques used in neural networks. A normalisation is done on the data to ensure that the data is not biased toward one of the features.

Using the principal component decomposition, the selected features are combined into a single feature which will be used to train the HMMs. This feature is shown in figure 5.12. The final correlation coefficient of the universal feature is -0.695 which is very close to that of best feature. Table 5.3 shows the total variance explained by each

principal component after the decomposition. It can be seen from this that the first principal component explains more than 70% of the variance. On page 31 the use of only this one principal component is mentioned. The classification system is to be trained on data from this figure. The "noisy" figure shows why a threshold method for classification will produce many false alarms. The "spikes" seem to jump arbitrarily and may trigger a "worn-tool alarm." There is fortunately a trend that can been seen in the data. The lower values of the dimensionally reduced feature vector correlates with worn-tool conditions. This figure also shows why an advanced classification system is needed for machining data.

Table 5.3: The principal components and the amount of the total variance the represent.

Principal component no.	Percentage of total variance
1	73.57
2	10.53
3	7.17
4	3.81
5	2.45
6	2.45
6	1.07
7	0.84
8	0.50
9	0.02

The first third of the data was selected to be the first class and the last third to be the second class. These two thirds of the data was used to calculate the P-vector for the dimensional reduction. These same thirds will be used in the next section for the training and testing.

The last preparation before the HMMs are applied is the discretisation. Because discrete HMMs are used, it is necessary to discretize the data. It was decided to quantise the data into 150 levels. At this level there is still ample detail left in the universal feature. A lower discretisation level will have sharper decision boundaries, this will probably give better classification results. More advanced HMM techniques however use continuous PDFs. Using a high number of discretisation levels therefore will give a better indication how future and more advanced models may perform. A better platform for comparison for the performance between discrete and continuous models is also created.

Figure 5.13 shows a few training sequences for sharp and worn tools. These figures were already discretized and represent the final product that is fed to the HMM models.

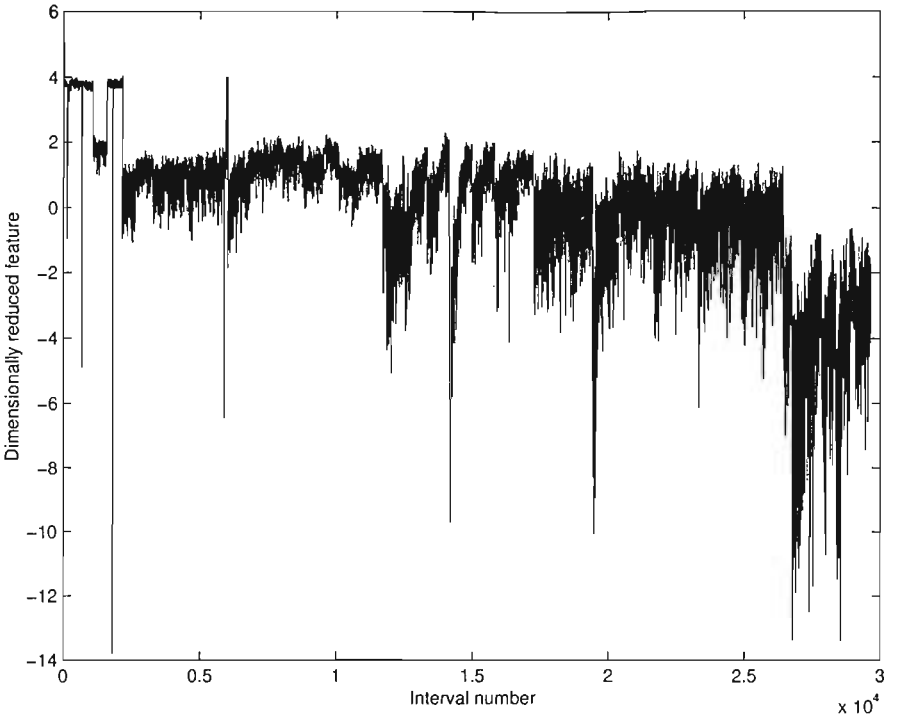

Figure 5.12: The final combined feature from which the training sequences for the HMM will be extracted.

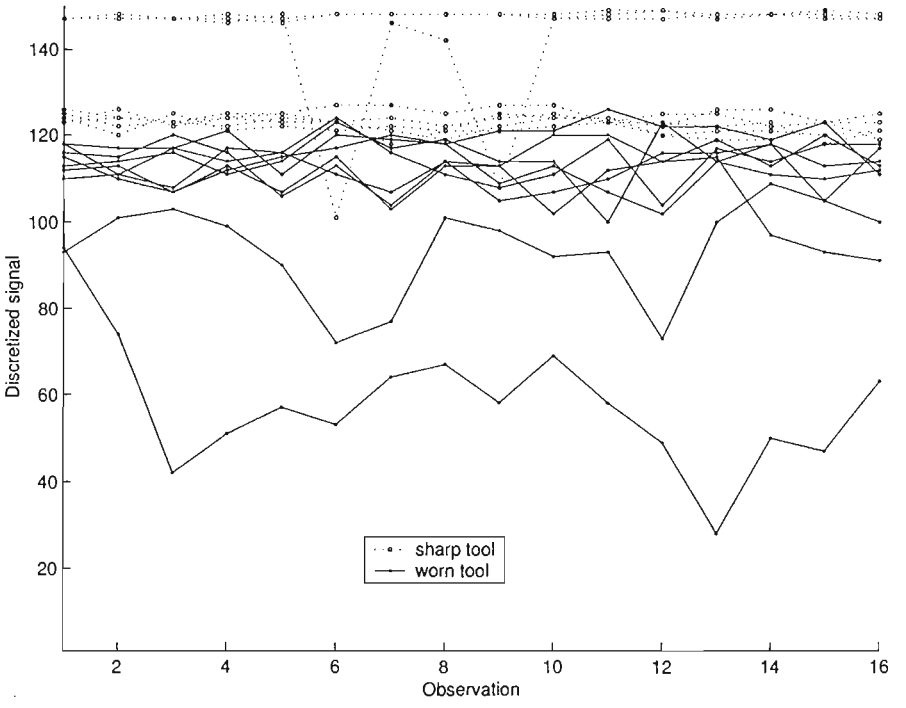

Figure 5.13: The training sequences after discretisation.

5.4 HMM training and classification

During the training, one HMM was trained for each of the identified classes. Each HMM was then trained on the data it is to be associated with. Afterwards the HMM were tested on mixed, unseen data. A HMM toolbox for MATLAB is used for the data classification techniques presented in this section.[1]

5.4.1 Selecting samples for training

Having already selected the classes to be recognised, it is necessary to select samples for training and for testing. From each class, one third of the samples are randomly selected and removed from the set. The remaining data is used for training. After training the models are tested with the remaining data. Figure 5.14 shows set of randomly selected training data samples. The different colours indicate different classes.

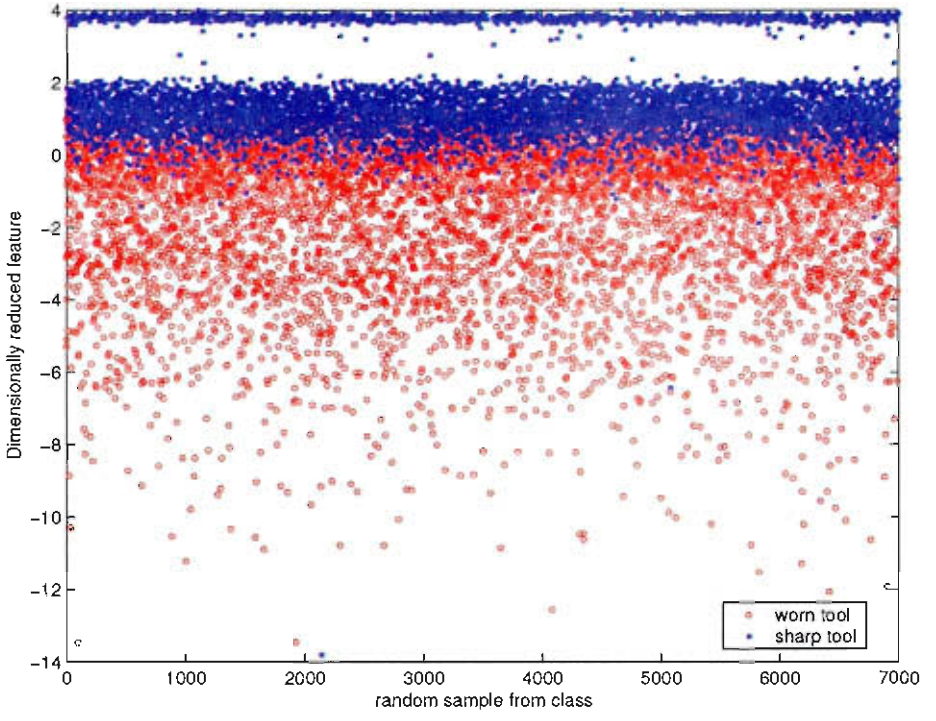

Figure 5.14: A training data set

The histograms in figure 5.15 show the areas of likelihood for samples in the different classes. Once again the tops of the bins were connected to form a curve. The second peak with the small variance in the histogram of the sharp tool, is an example of differences in workpiece composition and setup that affects the signal quality. From this figure it can

[1] Hidden Markov Model (HMM) Toolbox written by Kevin Murphy (1998). See http://www.ai.mit.edu/~murphyk/Software/hmm.html for details.

be seen that the PDFs of the sharp tools and the worn tools have a large overlap area. This will make recognition difficult.

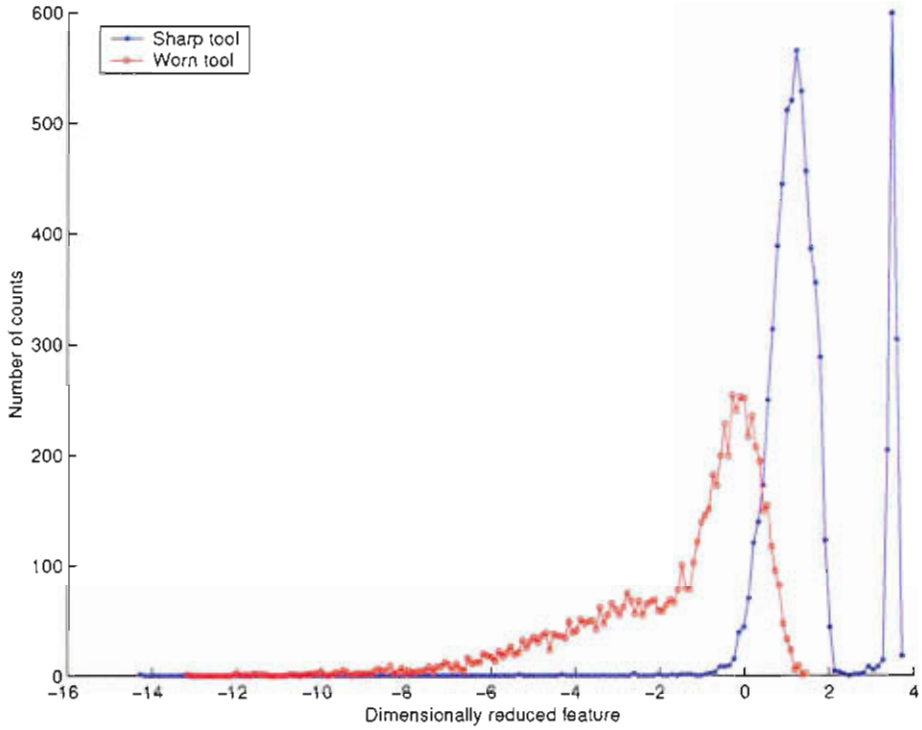

Figure 5.15: The histograms for the different classes

5.4.2 Condition for correct classification

At this point it is relevant to describe what is deemed to be a correct classification. After the HMM are trained they are tested. The testing entails that the HMMs are shown an unknown sequence. The sequences are similar to the ones shown in figure 5.13. The probability that each of the HMMs will produce the sequence is then calculated. The sequence is then classified in the class of the HMM with the highest probability. Since the testing data will have known labels (eg. the user has a prior knowledge of the class of the data), a correct classification will be when the HMM associated with the correct class has the highest probability.

5.4.3 The HMM topology

As with neural networks, there is no analytical way of predicting what HMM topology will produce the best results. An iterative procedure was followed where the whole training and testing procedure was repeated for an incrementally changing number of states. For each number of states, the training and testing was repeated five times and a mean was calculated. The results are shown in 5.16.

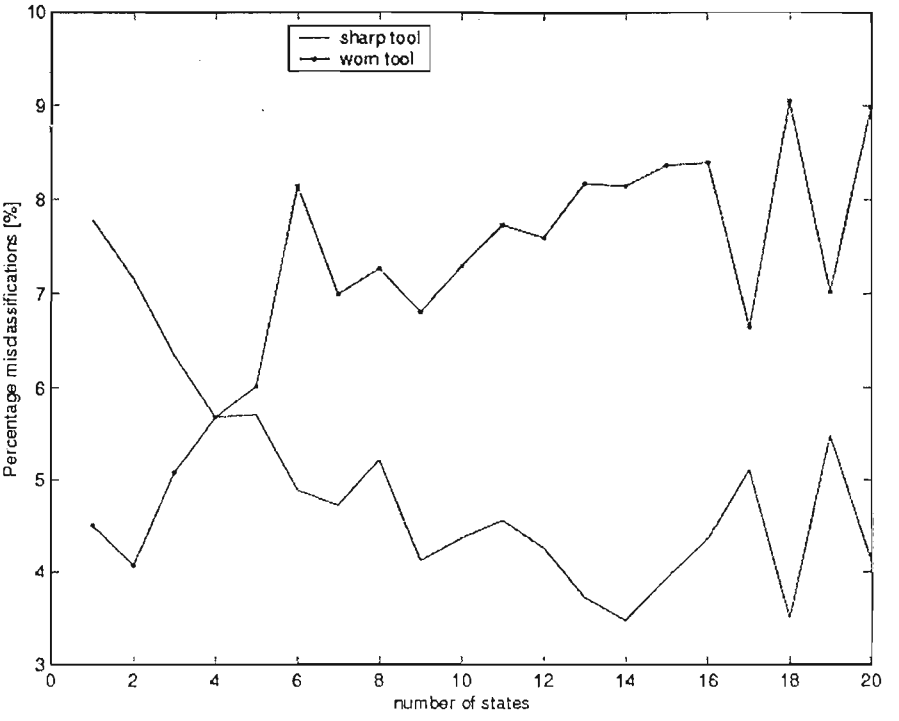

Figure 5.16: The number of states vs the recognition faults

From figure 5.16 the optimal number of states for each HMM with its associated class can be read off. This number will however be a trade off between complexity of the model and the performance. It was decided to select for:

- the HMM on worn tool data, the number of states as, 2
- the HMM on sharp tool data, the number of states as, 7.

With these parameters chosen, one can show more results of the HMM classification such as the forward probabilities.

5.4.4 Recognition and results

In order get an idea of the behaviour of the recognition of the HMMs, the test was repeated twenty with the chosen parameters. This is shown in figure 5.17. The mean of the performances are indicated on this figure. This figure shows that the behaviour is somewhat erratic, and can be ascribed to the quality of the data.

Figure 5.18 shows the probabilities that the HMMs will produce the testing data. The first half of the data is of class one and the second part is of class two. One then expects to see that the lines of the HMMs should cross in the middle somewhere. The probabilities of the HMMs are however a little more chaotic. The extreme dips in the data are caused by zeros in the probability density functions of the states. Since the data

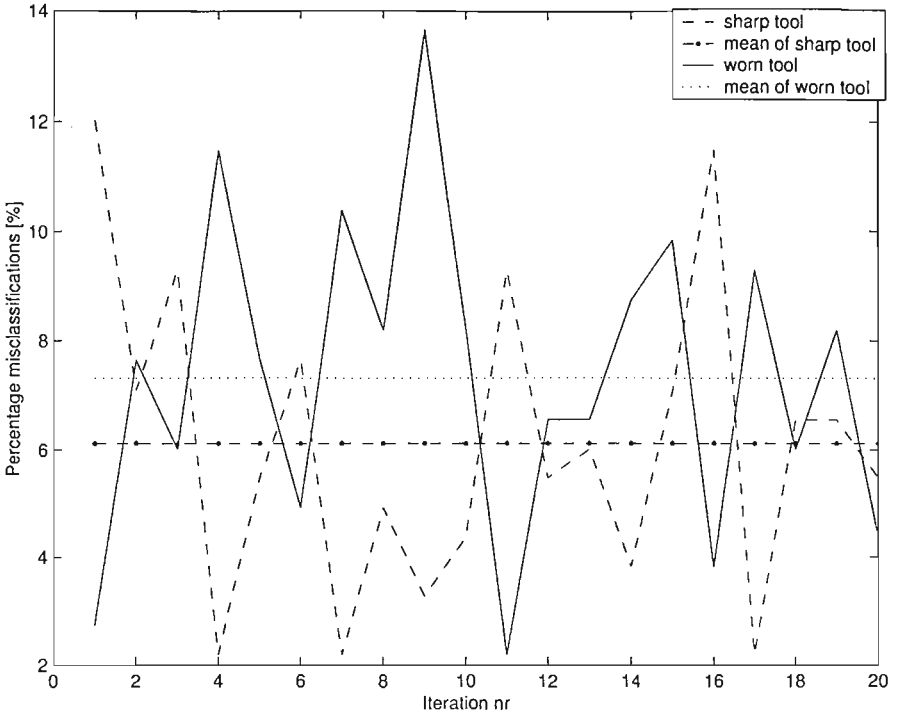

Figure 5.17: The behaviour of the classification performance.

is discretized it implies that the probability density functions are discretized as well. It may therefore happen that, in the PDF one of these "symbols" may have a probability of 0. The prediction probability of the HMM then drops to $-\infty$. A bound of -130 was put on this.

The outcome of this can be seen more clearly in figure 5.19. Once again in this figure, the data of the first class was shown in the first half and that second class in the second half. Correct classifications are therefore shown as red circles in the first half and the blue circles in the second half. On average this quantifies into:

- 6.5% incorrect classifications of sharp tools
- 7.5% incorrect classifications of worn tools

5.5 The Maximum Likelihood classifier

To create a basis for comparison, the recognition was also to be done with another maximum likelihood technique. The maximum likelihood was chosen for this. This type of classifier is easy to implement and usually very robust. The maximum likelihood classifier works by creating a decision boundary using the PDFs of the different classes. The training data in this case is used to fit Gaussian PDFs onto the data. Because of the smooth decision boundaries that this method creates, it is easy to predict the behaviour

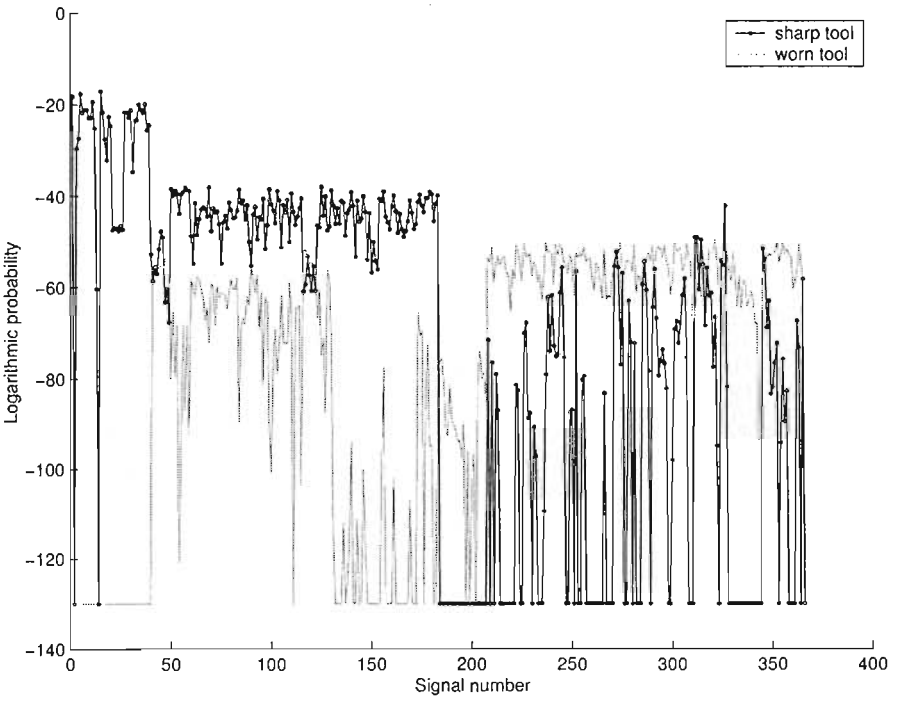

Figure 5.18: The prediction probabilities of the HMMs

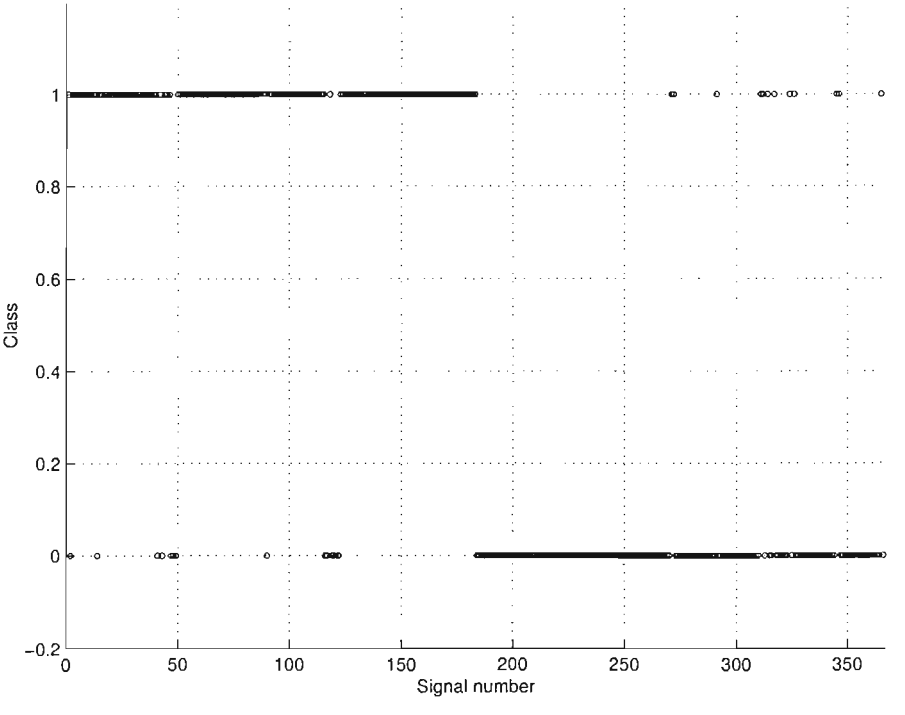

Figure 5.19: The classification results

of the system. The performance of this system is dependant on the quality of the fit of the Gaussian PDFs on the data classes.

Formally the maximum likelihood classifier works as follows: if $P_a(x)$ is the PDF of class a and similarly, $P_b(x)$ is the PDF for class b, then the decision boundary will be where:

$$P_a(x) - P_b(x) = 0 \tag{5.2}$$

Classification can then be done on any arbitrary value of x. If equation 5.2 is calculated for a value of x and the answer is a positive number, then x belongs the class a, otherwise class b. Figure 5.20 shows the PDFs for the two classes and the decision boundary.

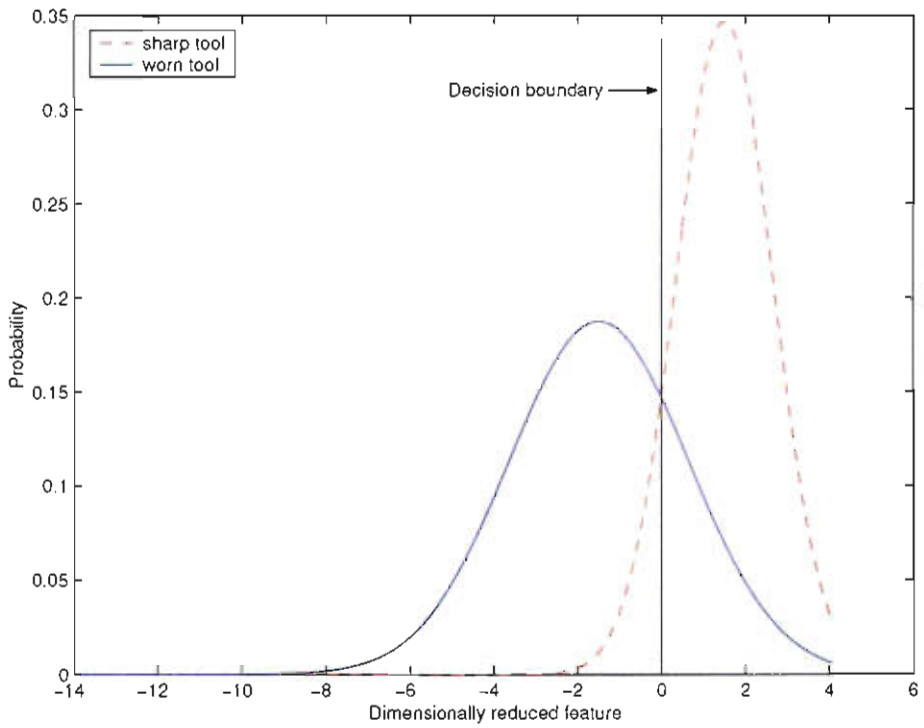

Figure 5.20: The Gaussian PDFs fitted onto the data and the decision boundary.

The training of the maximum likelihood classifier is done by simply drawing the histograms from a training data set with the defined classes. The decision boundary is then applied to a testing set. An example of this decision boundary, plotted on a training set in figure 5.21.

The training and testing is repeated a number of times and the performance and the behaviour of the performance is shown in figure 5.22.

From this figure, it can be seen that the performance is rather stable. The average performance for the maximum likelihood classifier turns out to be:

- $3,2\%$ incorrect classifications for a sharp tool
- $27,3\%$ incorrect classifications for a worn tool

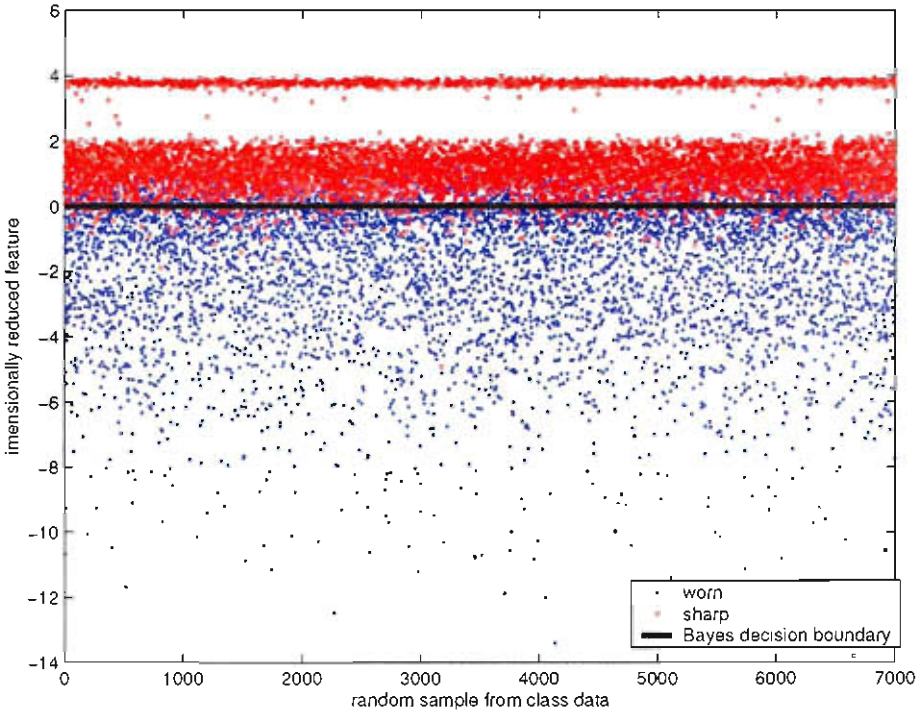

Figure 5.21: The training data with the decision boundary applied.

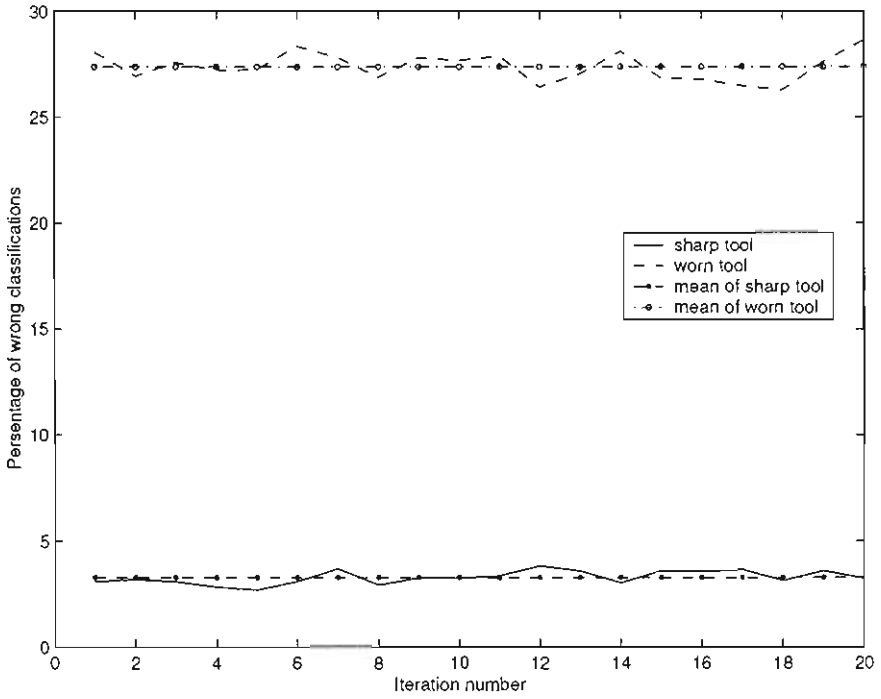

Figure 5.22: The performance of the maximum likelihood classifier over a number of iterations.

The low classification result for worn tool shows that a Gaussian PDF is not a very good approximation for the likelihood function of the worn tool. The reason for the low result is the high amount of overlap between the two PDFs, which is clear in figure 5.20. In figure 5.15 it can be seen that there is an amount of overlap between the histograms. This overlap is a lack of separation between classes. The observation sequences that the HMMs use is one way to overcome this problem because temporal characteristics are taken into account. Comparison between the two methods of classification shows that the HMM classification is not as influenced by this lack of separation.

5.6 Reduced dataset

The performance of any classification algorithm is dependant on the quality and the quantity of the data that is used to train the system. It has been argued that the performance of the HMM recognition system will improve if there is more data and better quality data available for training. In order to show that this was the case, the data set was reduced and the training and testing of the HMM classification system was repeated.

For this trail the first 75% of the data was used. Again the data was divided into three classes of which the last and first were used in the classification tests. Again two thirds of the data of each class were randomly selected to train the system and the last third was used to test the system.

A principal component decomposition was once again applied to the two classes to achieve separation and dimensional reduction. After the dimensional reduction, two histograms were drawn up of each class. This is shown is figure 5.23. It can be seen that less prominent separation is achieved between the two classes.

Once again the whole classification procedure was done exhaustively to find the "optimal" number of states for this application. According to figure 5.24 the optimal for this case is:

- 2 states for the worn tool data
- 8 states for the sharp tool data

It is evident that there is much less of a trend between classification performance and the number of states of the HMMs. Each data point on the graph represents the mean of an average of 5 classification iterations.

When these figures for optimal classification are applied for investigation into the behaviour of the classification test results, figure 5.25 is the result. As with the previous result, the performance behaviour of the HMM classification is rather erratic, much more than with the full data set.

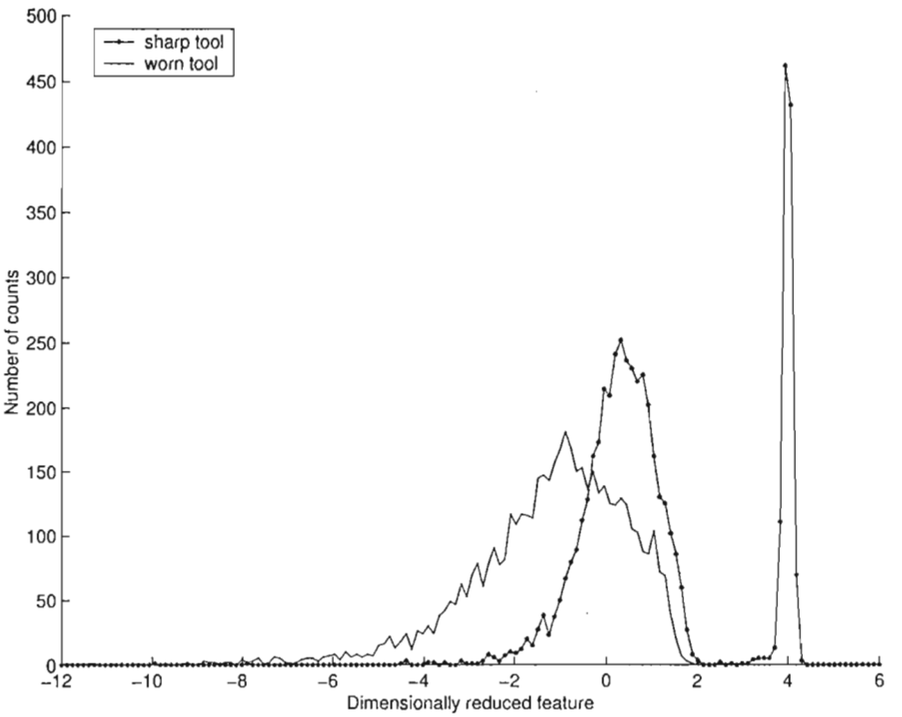

Figure 5.23: The histogram of the two classes in the reduced data set.

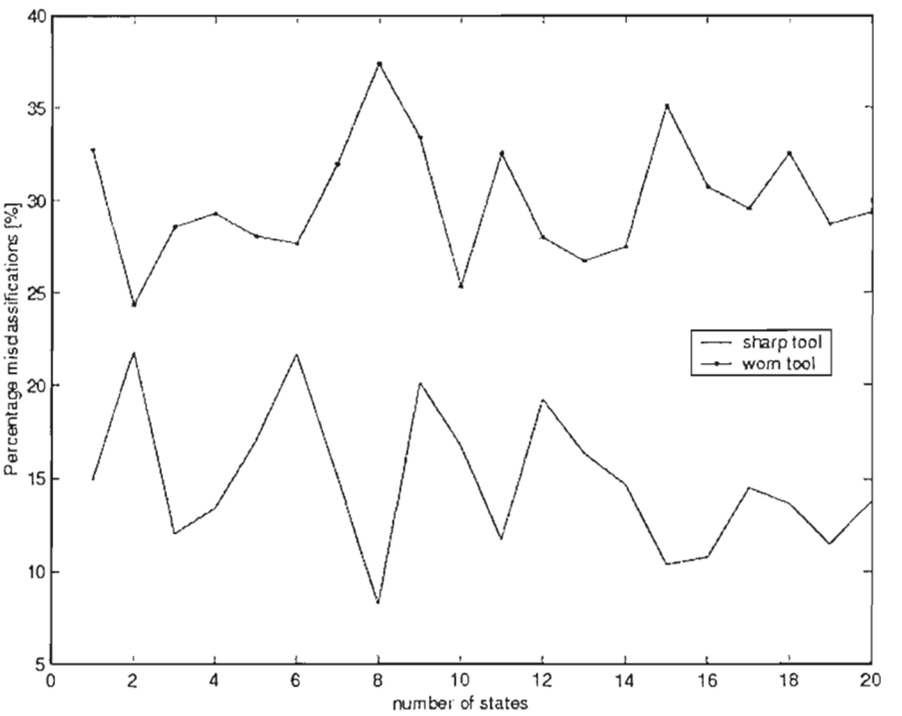

Figure 5.24: The classification performance as a function of the number of states.

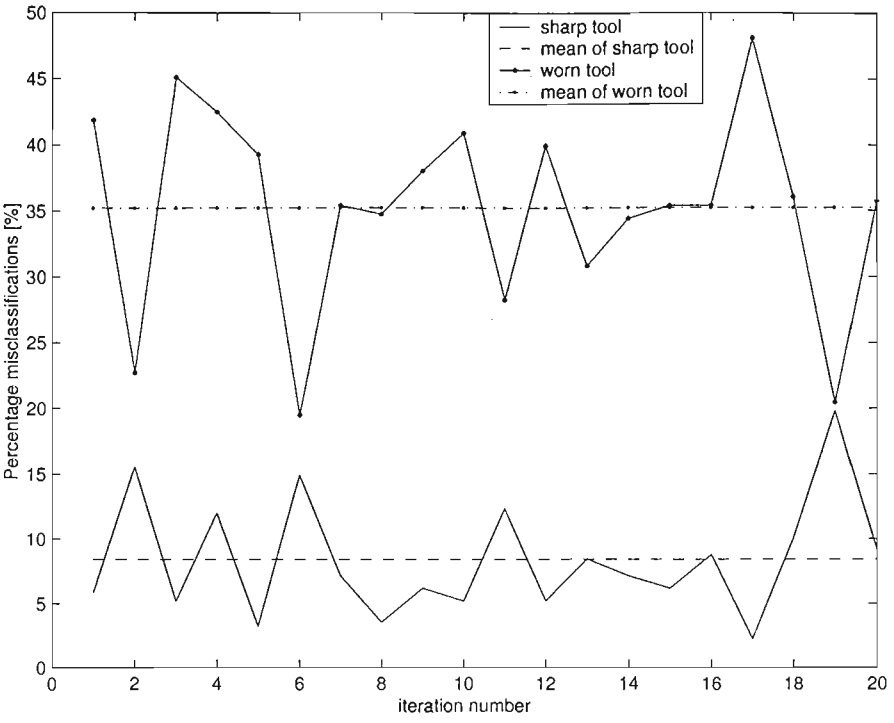

Figure 5.25: The behaviour of the classifications.

The average result of the classification for the reduced data set was:

- 35% incorrect classifications for worn tools
- 8% incorrect classifications for sharp tools

From this it shows that with less data there is lack of separation between classes which hampers the performance of the HMM classifier. This also validates the claim that more data will enable more, and better class allocation and separation, and ultimately better classification