

Automatic Recognition of Machinery Noise in the Working Environment

Primož Lipar* – Mirko Čudina – Peter Šteblaj – Jurij Prezelj
University of Ljubljana, Faculty of Mechanical Engineering, Slovenia

A necessity for the suitable recognition of different machinery and equipment based on the sound they generate is constantly present and will increase in the future. The main motivation for the discrimination between different types of machinery sounds is to develop algorithms that can be used not only for final quality inspection but for the monitoring of the whole production line. The objective of our study is to recognize the operation of the individual machine in a production hall, where background noise level is high and constantly changing. An experimental plan was designed and performed in order to confirm the hypothesis proposing that automatic speech recognition algorithms can be applied to automatic machine recognition. The design of the automatic machine recognition procedure used in our study was divided into three stages: feature extraction, training, and recognition (classification). Additionally, a traditional mel-frequency cepstral coefficient (MFCC) procedure was adjusted for machinery noise by using different filter compositions. Finally, two classifiers were compared, the k-NN classifier and the multivariate Gaussian distribution. The results of the experiment show that machinery noise features frequency cepstral coefficients (FCC) should be extracted by using linear filter compositions and processed with recognition algorithm based on the multivariate Gaussian distribution.

Keywords: machinery noise, machinery classification, k-NN classifier, multivariate Gaussian distribution classifier, frequency cepstral coefficients

Highlights

- Implementation of automatic speech recognition algorithms for machinery noise classification.
- Optimization of MFCC filter banks for machinery noise.
- Application of multivariate Gaussian distribution as statistical modelling of machinery noise.
- Comparison of multivariate Gaussian distribution classifier with k-NN classifier.
- Experimental validation of proposed approach.

0 INTRODUCTION

Automation of production started in early 1913 when Henry Ford invented the conveyor belt. Automation remains one of the main topics of engineering because it offers many benefits; it increases product quality due to the reduction of human error; it increases production volume, it protects human health via the automation of dangerous tasks, and ultimately any threat to life can be minimized. The automation of any process inherently depends on the feedback information from the process itself. Feedback information can be obtained by acquiring different signals from within the controlled system. The sound generated by the machinery can be used for feedback information as reported in numerous examples; Mechanical and electrical faults observed on induction motors have been classified using analysis of the acoustic data by using correlation and wavelet-based analyses for extracting necessary features from the acoustic data [1]. A new technique of acoustic-based diagnosis (ABD) for gearboxes based on near-field acoustic holography (NAH) and spatial distribution features of the sound field was presented in [2]. The fault diagnosis of bearings has been exploited through

a machine learning approach by using acoustic signals acquired from the near field area of bearings in good and simulated faults. The descriptive statistical features were extracted from sound signals, and the important ones were selected using a decision tree (dimensionality reduction) [3]. Vacuum cleaner motors quality end-test at the end of the manufacturing cycle have been presented in [4]. Efficient signal processing algorithms have been developed to detect and localize bearing faults, defects in fan impellers, improper brush-commutator contacts and the rubbing of rotating surfaces [4].

Sound is typically used for the end quality inspection of produced machinery. Experts learn with experience to listen to the sound of the operating machinery, and use it as an important source of information for the classification between flawless and faulty machines. However, human responsiveness and accuracy are sometimes insufficient for the real-time detection of a failure on produced machinery in mass production. Therefore, the need for suitable recognition of sound generated by the machinery is constantly present and will certainly increase in the future. The improvement of the recognition algorithms

can further be used not only for end quality inspection but for monitoring the whole production line.

Recognition of different sounds has a long history, with its roots in 1952 when Bell Labs demonstrated the first automatic speech recognition (ASR) systems for small-vocabulary recognition of digits spoken over the telephone. After computers had grown in power during the 1960s, filter banks were combined with dynamic programming to produce the first practical recognizers, mostly for words spoken in isolation. In 1970 linear predictive coding (LPC) became a dominant ASR algorithm. During the 1980s, it was replaced with mel-frequency cepstral coefficients (MFCCs). During the 1990s, commercial applications evolved from isolated-word dictation systems to general-purpose continuous speech systems [5] and [6].

Since the mid-1990s, ASR has been largely implemented in commercial software. Medical reporting and legal dictation have been two applications driving the development of ASR, as well as the automation of services to the public over the telephone [5]. Applications of ASR methods were extended to environmental sound recognition in 1993. The MFCC parameterization method was implemented in combination with the class statistic classifier [7]. Significant progress on environmental sound recognition was achieved when MFCC parameterization was combined with the matching pursuit (MP) algorithm [8]. MFCC has been widely used in environmental sound recognition as well as its first and second derivatives in combination with other parameterization methods [9] and [10]. Available literature studies about industrial machine sound recognition are usually based on the Morlet wavelet parameterization approach [11] to [13]. Furthermore, MFCC can be used as a damage-sensing feature because its compactness and decorrelation characteristics make it particularly suited for statistical recognition applications. Attempts to use the MFCC parameterization method to extract features of industrial sounds to detect faults was already discussed in the literature [14] and [15], with no modification to the algorithms.

The objective of our study is to recognize the operation of an individual machine in a production hall where the background noise level is high and constantly changing. The knowledge transfer from ASR to machinery sound identification seems to be a reasonable choice due to the efficiency of speech and speaker identification. Human speech is more dynamic than the sound produced by machines. The time stability of non-natural sounds should even

increase the identification efficiency. The purpose of this paper is to show that the pattern recognition and feature extraction methods used in ASR systems, after minor modifications, can be efficiently used to identify different machine sounds even in noisy environments. An experimental plan was designed and performed in order to confirm the hypothesis proposing that ASR algorithms can be applied for automatic machine recognition (AMR). The design of the AMR procedure used in our study was divided into three stages: feature extraction, training, and recognition (classification).

The procedure presented in this paper is used in the system for the automatic classification of noise events during environmental noise measurements [16]. The same procedure was also tested for detection of cavitation phenomena on kinetic pumps [17]. The influence of various rotor designs on the generated noise is also tested with the same procedure.

1 FEATURE EXTRACTION, TRAINING, AND RECOGNITION

Feature extraction is essential in the ASR and AMR due to the quality of training models. Likewise, pattern matching strongly depends on the quality of feature extraction methods. There are many different features of sound signals:

- **Temporal features** are computed from the waveform or the signal energy envelope: log-attack time, temporal decrease, temporal centroid, effective duration, zero-crossing rate, cross-correlation.
- **Energy features** refer to various energy contents of the signal: global energy, harmonic energy, and noise energy.
- **Spectral features** are computed from the short-time Fourier transform (STFT) of the signal: spectral centroid, spread, skewness, kurtosis, slope, decrease, roll-off point, variation, etc.
- **Harmonic features** are computed from the sinusoidal harmonic modelling of the signal: fundamental frequency, noisiness, odd-to-even harmonic ratio, tristimulus, deviation, centroid, spread, skewness, kurtosis, slope, decrease, roll-off point, variation.
- **Perceptual features** are computed using a model of the human hearing process: MFCC, first order derivative of MFCC (DMFCC), and second order derivative of MFCC (DDMFCC), loudness, specific loudness, sharpness, spread, roughness, and tonality [18].

To extract sound features from the sound signal with dynamic level changes, similar to speech, more

complex parameterization methods have to be used, as discussed in publications [19] and [20]. Most commonly used parameterization method in speech or speaker recognition systems is the MFCC method. It was used by many authors and, according to the literature survey, gives the best classification results in ASR. The time constant of features extraction is essential. In our application it turned out that the noise generated by the observed machinery is less dynamic than speech; therefore, the length of time window was set to 50 ms. Machinery sounds are usually generated by rotational movements that cause a harmonic form of noise. Machines have different sizes, operational conditions (speed and load), tasks, etc., thus leading to the different distribution of spectral energies. Frequency cepstral coefficient (FCC) parameterization is based on the deconvolution of spectral energies obtained from certain frequency ranges, thus making it a reasonable choice for machinery sound recognition.

1.1 Feature Extraction Based on the FCC Parameterization

The performance of FCC may be affected by the number of filters, the shape of filters, the way that filters are spaced and the way that the power spectrum is warped [21]. A chart diagram of FCC parameterization, as used in our experimental application, is shown in Fig. 1.

The basic FCC feature extraction procedure was performed by applying a standard mel filter, and

additionally by applying two different filter scalings: linear and logarithmic.

To calculate the FCC, the signals are first divided into short time intervals. Hamming windowing is then applied and fast Fourier transformation (FFT) of each time window for the discrete-time signal $x(n)$ with length N is calculated. The magnitude spectrum $|X(k)|$ is now scaled in both frequency and magnitude, where k stands for the FFT index of frequency. First, the frequency is scaled using one of the filter banks $|H(k, m)|$. Then the logarithm is taken, according to Eq. (1):

$$X'(m) = \ln \left(\sum_{k=0}^{N-1} |X(k)| H(k, m) \right). \quad (1)$$

For $m=1, 2, \dots, M$, where M is the number of filters in a bank. Three different filter banks were used in our application; a mel scale filter bank, a linear filter bank, and a logarithmic filter bank. All three filter banks are based on a collection of triangular filters defined by the centre frequencies $f_c(m)$, as written in Eq. (2). Central frequencies are given in Eqs. (5), (6) and (10) for mel scale, linear scale and logarithmic scale respectively.

$$H(k, m) = \begin{cases} 0 & \text{for } f(k) < f_c(m-1) \\ \frac{f(k) - f_c(m-1)}{f_c(m) - f_c(m-1)} & \text{for } f_c(m-1) \leq f(k) < f_c(m) \\ \frac{f_c(m+1) - f(k)}{f_c(m+1) - f_c(m)} & \text{for } f_c(m) \leq f(k) < f_c(m+1) \\ 0 & \text{for } f(k) \geq f_c(m+1) \end{cases} \quad (2)$$

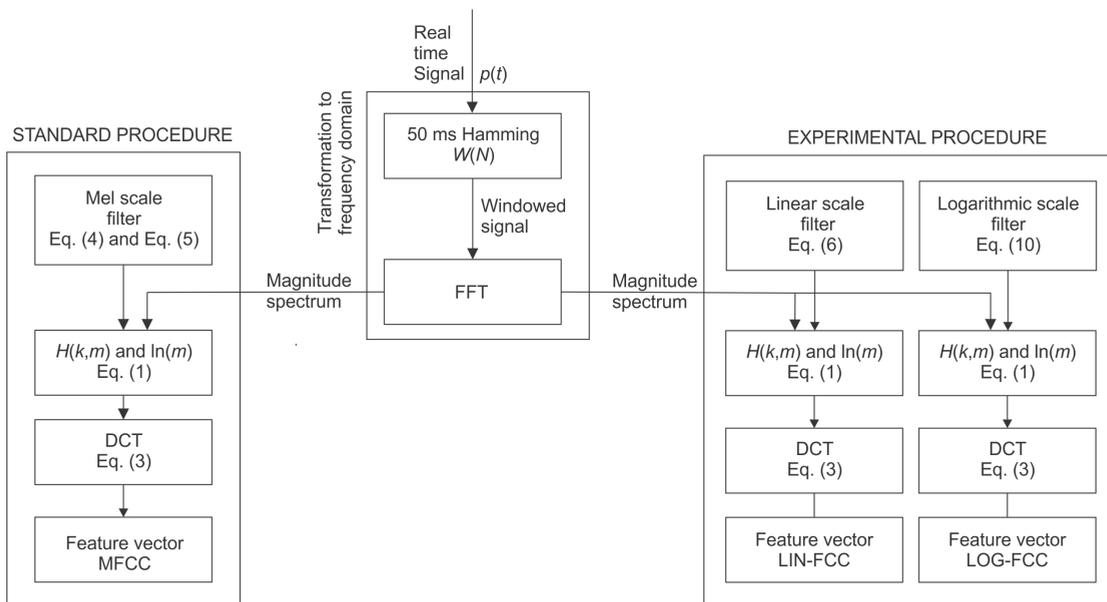


Fig. 1. Chart diagram of FCC parameterization

Three different types of FCCs (MFCC, lin-FCC and log-FCC) are obtained from filtered $X'(m)$ using Eq. (3):

$$c(l) = \sum_{m=1}^M X'(m) \cos(l \frac{\pi}{M} (m - \frac{1}{2})). \quad (3)$$

For $l=1, 2, \dots, 12$, where $c(l)$ is the l th FCC [22]. Three different types of filters influence the $c(l)$ through the filtered magnitude spectre $X'(m)$.

1.1.1 Mel Scale Filter

MFCCs are obtained from $X(m)$ which is filtered with a mel-scale filter bank. The centre frequencies of filters in the mel filter bank are computed by approximating the mel scale with:

$$\phi = 2595 \log_{10} \left(\frac{f}{700} + 1 \right), \quad (4)$$

which is a common approximation. Afterward, the fixed frequency resolution in mel scale is computed, corresponding to the logarithmic scaling of the repetition frequency, using $\Delta\phi = (\phi_{\max} - \phi_{\min}) / (M+1)$. Therefore, ϕ_{\max} is the highest frequency of the filter bank on the mel scale, computed from f_{\max} using Eq. (4), ϕ_{\min} is the lowest frequency in mel scale, and M is the number of filters in a bank. The centre frequencies on the mel scale are given by $\phi_c(m) = m\Delta\phi$ for $m=1, 2, \dots, M$ [22]. By using the inverse of Eq. (4), centre frequencies in Hz are obtained by:

$$f_c(m) = 700(10^{\phi_c(m)/2595} - 1). \quad (5)$$

The low-frequency limit and high-frequency limit of the individual filter in a bank are obtained by taking the central frequency of the previous and next filter. Because mel scale is adjusted to human speech and hearing, it provides good results in speaker/speech identification systems. Due to the temporal stability and linearity of machine sounds, it was essential to test the linear and the octave-based frequency scaling to calculate lin-FCCs and log-FCCs.

1.1.2 Linear Scale Filter

Central frequencies of the linear filter bank are obtained with the linear equation:

$$f_l(m) = f_{\min} \cdot \Delta f \cdot m, \quad (6)$$

for $m=1, 2, \dots, M$, where

$$\Delta f = \frac{f_{\max} - f_{\min}}{(M+1)}. \quad (7)$$

Low filter frequencies are computed by using:

$$f_l(m) = f_{\min} \cdot \Delta f \cdot m, \quad (8)$$

for $m=0, 1, \dots, (M-1)$, and high filter frequencies are computed by using:

$$f_h(m) = f_{\min} \cdot \Delta f \cdot m, \quad (9)$$

for $m=2, 3, \dots, (M+1)$.

1.1.3 Logarithmic Scale Filter

Comparison of the results obtained by mel and linear filters with 1/n octave filters is not possible, because of the different number of filters in the collection. Therefore, logarithmic composition was used, due to its similarity with the central frequencies composition of the octave band filters. The central frequencies of such filters were calculated using this equation:

$$f_c(m) = 10^{m\Delta f_{\log} + \log_{10}(f_{\min})}, \quad (10)$$

for $m=1, 2, \dots, M$, where Δf_{\log} is calculated by:

$$\Delta f_{\log} = \frac{\log_{10}(f_{\max}) - \log_{10}(f_{\min})}{M+1}. \quad (11)$$

Low frequencies of filters are calculated by:

$$f_l(m) = 10^{m\Delta f_{\log} + \log_{10}(f_{\min})}, \quad (12)$$

for $m=0, 1, \dots, (M-1)$, and high filter frequencies by:

$$f_h(m) = 10^{m\Delta f_{\log} + \log_{10}(f_{\min})}, \quad (13)$$

for $m=2, 3, \dots, (M+1)$.

1.2 Classification Using k-NN and MGD

The classification process of machinery into classes, based on generated sounds, is divided into two steps: training and testing. To train the algorithm and obtain the reference template, enough samples of generated sound should be available for each class. Extracted features from sound signals generated by known sources are used during the training procedure when the class reference template model is learning. In the second step, the sound signal from the unknown machine is matched with a stored reference template model, and classification decisions are made [23]. Two classification algorithms, which have diametrically opposed classification approaches, were tested: k -nearest neighbour (k -NN) and multivariate Gaussian distribution (MGD). MGD can be regarded as a single state Gauss Mixture Model (GMM). Simpler MGD was selected over the GMM, because it provides better results when adapting new data into the database [24].

1.2.1 *k*-NN Algorithm

The *k*-NN algorithm is one of prospective statistical classification algorithms used for classifying objects based on the closest training examples in the feature space. It is a lazy learning algorithm in which the *k*-NN function is approximated locally, and all computations are deferred until classification. No actual model or learning is performed during the training phase, although a training dataset is required. It is used solely to populate a sample of the search space with instances whose class is known. For this reason, this algorithm is also known as the lazy learning algorithm, which means that the training data points are not used to do any generalization, although all the training data is needed during the testing phase.

One of the advantages of the *k*-NN method in classifying the objects is that it requires only a few parameters to tune: *k* and the distance matrices, for achieving sufficiently high classification accuracy. Thus, in *k*-NN-based implementation, the best choice of *k* and distance metric for computing the nearest distance is a critical task. Generally, large values of *k* reduce the effect of noise on the classification but make boundaries between classes less distinct [25].

k-NN classification is based on measuring the Euclidean distance (Eq. (14)) between unknown sample vector \mathbf{X}_s and known sample vectors $\mathbf{X}_m = [X_1, X_2, \dots, X_m]$ sorted in the database.

$$d_{sm} = \sqrt{\sum_{n=1}^N (x_{sn} - x_{mn})^2}, \quad (14)$$

where index *n* is the size of observations and *m* is an index of the individual vector.

1.2.2 Multivariate Gaussian Distribution

The multivariate Gaussian distribution (MGD) is a generalization to two or more dimensions of the univariate Gaussian (or normal) distribution, which is often characterized by its resemblance to the bell shape. This algorithm is selected according to the statistical analysis of the reference feature vectors. The histogram of individual feature vector components indicates their Gaussian distribution. An additional advantage of the algorithm is its high computational speed. MGD is the Gaussian mixture model with a hypothetical mixture.

During preliminary studies individual vector components were identified to have the univariate Gaussian distribution with acceptable deviation. If the real-valued univariate random variable *X* is said

to have the Gaussian distribution with mean μ and variance σ^2 (written as $X \sim N(\mu, \sigma^2)$), then its density function can be written as:

$$f(x | \mu, \sigma) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}, \quad (15)$$

where $-\infty < \mu < \infty$ and $\sigma > 0$. The vector-valued random variable $\mathbf{X} = [X_1, \dots, X_r]^T$ is said to have multivariate Gaussian distribution with mean *r*-vector $\boldsymbol{\mu}$ and positive defined symmetric (*r* × *r*) covariance matrix Σ , if its density function is given by the curve;

$$f(\mathbf{x} | \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{-1/2} |\Sigma|^{-1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})}, \quad (16)$$

$\mathbf{x} \in \mathfrak{R}^r$. If \mathbf{X} is a random *r*-vector with values within \mathfrak{R}^r , then its expected value is the *r*-vector:

$$\boldsymbol{\mu}_x = E(\mathbf{X}) = [E(X_1), \dots, E(X_r)]^T = [\mu_1, \dots, \mu_r]^T, \quad (17)$$

and the (*r* × *r*) covariance matrix of \mathbf{X} is given by:

$$\begin{aligned} \Sigma_{xx} = \text{cov}(\mathbf{X}, \mathbf{X}) &= E[(\mathbf{X} - \boldsymbol{\mu}_x)(\mathbf{X} - \boldsymbol{\mu}_x)^T] = \\ &= E[(X_1 - \mu_1, \dots, X_r - \mu_r)(X_1 - \mu_1, \dots, X_r - \mu_r)^T] = \\ &= \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \dots & \sigma_{1r} \\ \vdots & \ddots & \ddots & \vdots \\ \sigma_{1r} & \sigma_{r2} & \dots & \sigma_r^2 \end{pmatrix}, \end{aligned} \quad (18)$$

where

$$\sigma^2 = \text{var}(X_i) = E[(X_i - \mu_i)^2], \quad (19)$$

is the variance of X_i , $i = 1, 2, \dots, r$, and

$$\sigma_{ij} = \text{cov}(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)], \quad (20)$$

is the covariance between X_i and X_j , $i, j = 1, 2, \dots, r$ ($i \neq j$) [26].

A covariance matrix (also known as dispersion matrix or variance-covariance matrix) is a matrix whose element in the *i*, *j* position is the covariance between the *i*th and *j*th elements of a random vector. Each element of a vector is a scalar random variable, either with a finite number of observed empirical values, or with a finite or infinite number of potential values specified by a theoretical joint probability distribution of all the random variables. Multivariate Gaussian distribution density exists when the symmetric covariance matrix is positive definite. A sample covariance can be calculated using following equation:

$$\sigma_{ij} = \frac{1}{N-1} \sum_{n=1}^N (x_{ni} - \mu_i)(x_{nj} - \mu_j), \quad (21)$$

where N is the number of samples of a X_i or X_j vector component.

2 EXPERIMENT

An experimental plan was designed to evaluate the hypothesis proposing that an ASR algorithm can be applied for AMR. A traditional FCC method from ASR was used to extract sound features from typical industrial and machinery sounds. The classification of feature vectors was performed by k -NN and MGD at different signal-to-noise ratios (SNRs). The industrial noise was mixed with machinery sounds as background noise to set the SNRs. All machinery signals were first pre-processed to assure equal energies. SNR was set to 7 dB and 2 dB below the signal level by mixing industrial noise having appropriate energy levels with sample signals. Five different machines and three different operations were included in the MSR. The classification performance was tested by applying different filter types and different numbers of filters to FCC calculations. Experiments were conducted for traditional mel scale filter banks, and for two additional filter banks: the Linear and Logarithmic filter banks. The experimental plan which was designed and performed is presented in Table 1.

Measurements of machinery sounds were performed in a laboratory environment. The experimental set-up is shown in Fig. 2, where the position of measured machines and microphone are constrained, as well as room dimensions. To test the robustness of the proposed algorithms, original sounds were mixed with background noise having three different sound pressure levels (SPLs). All machines sound SPLMs were set to equal levels. Background noise SPLBG was adapted to ensure three different SNRs. The reverberation time of the room was 0.8 s. Sounds of two different vacuum cleaners, jigsaw, compressor, and drill were recorded using a B&K microphone and M-audio sound card. Also, the sounds of the jigsaw cutting aluminium plate, drilling holes into a steel plate and grinding

aluminium plate on a grinding machine were recorded to simulate different operations. One-minute sound samples were recorded without background noise and with a sampling frequency of 48000 Hz and 16-bit resolution. Background noise was recorded in the industrial hall to simulate a real industrial environment. Parameterization of sound samples and classification was done in a program written in LabVIEW. The program was designed to allow the user to change FCC parameters (time window length, overlapping, frequency range, filter type, and the number of filters). FCCs were extracted to 50 ms time windows of the signal with 40 % overlapping in a frequency range from 100 Hz to 8000 Hz.

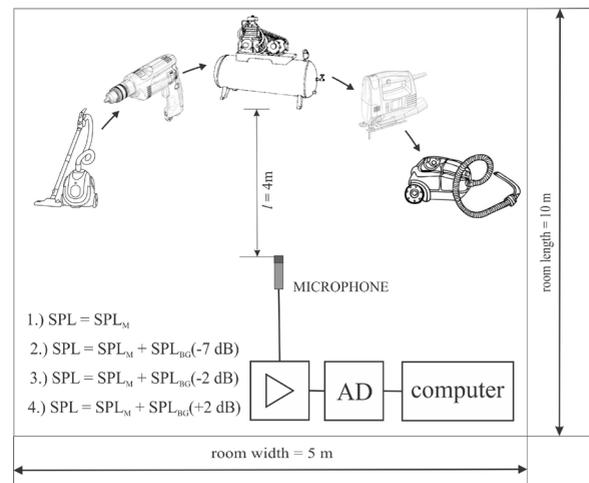


Fig. 2. Experiment set-up scheme

2.1 Database Design

Because the k -NN algorithm has no training mode, its database is just a compilation of vectors composed from FCCs, which result from pure machinery sound signals. In contrast, the MGD algorithm works only with the reference template models. FCCs of sound samples with no background noise were used in a training mode to build a MGD reference template model. The database for the MGD classifier

Table 1. Parameters of experimental plan

Parameter	1	2	3	4	5	6	7
Signal / noise	7 dB	2 dB	-2 dB	/	/	/	/
Filter size	3	6	12	18	24	30	36
Algorithm	k-NN	MGD	/	/	/	/	/
Filter type	mel	linear	Log	/	/	/	/
Machine type	compressor	vacuum cleaner 1	Jigsaw	drill	vacuum cleaner 2	/	/
Machinery operations	jigsaw cutting Al	drilling	Grinding	/	/	/	/

parameters is composed of mean value vector $\boldsymbol{\mu}$, the determinant of covariance matrix $|\Sigma|$ and inverse of covariance matrix Σ^{-1} .

If the covariance matrix is not full rank, then the multivariate normal distribution is degenerate and does not have a density. More precisely, it does not have a density with respect to the r -dimensional Lebesgue measure. In other words, the r dimensional MGD of vector \mathbf{X} has density if the covariance matrix is positive definite i.e. feature vector components are independent. If non-diagonal covariances are zero valued, then the covariance matrix becomes a diagonal variance matrix.

In practice, the non-diagonal covariances of vector components are not necessarily zero valued. This can result in a negative matrix determinant. If its determinant is zero, the matrix becomes non-invertible. Eq. (16) indicates that the determinant of a covariance matrix must be positive. It is positive in the case of a diagonal variance matrix but not necessarily otherwise. Individual components of vector \mathbf{X} have 12 coordinates (12 FCCs). An individual vector component was treated as the normally distributed univariate. If the 12 dimensional covariance matrix of vector \mathbf{X} is positive definite than MGD density exists, otherwise we must reduce components to the number at which point the covariance matrix become positive definite. The MGD database size of the individual model depends on the size of a positively definite covariance matrix.

2.2 Classification

The MGD rate of machinery sound recognition was determined by applying the voting procedure, where the decision of affiliation is based on a maximum matching of the unknown feature vector with the

reference template models. MGD was used as a parametric model of the probability distribution for FCC feature measurements. The MGD algorithm calculates the probability of affiliation for an unknown feature vector based on the stored models and votes for the one with the maximum probability. The k -NN algorithm is quite different from MGD. It simply measures the Euclidian distances between the unknown feature vector and those stored in the database and votes for the k minimum measured distances.

The experimental design included the analysis of both, the impact of the number of filters and different composition of filters on a classification performance. The experimental plan also compared the classification performance of two classifiers. Classification performances have been tested for two different SNRs to simulate real environmental conditions, where performance can be degraded due to all sorts of disturbances.

3 RESULTS AND DISCUSSION

Classification performance results from the k -NN and MGD classifiers for a different number of filters in the filter bank are shown in Table 2. The results are presented as an average of classification performance of five different machine sounds. Classification performance for 12 filters gives the best performance for all filter types, regardless of SNR. The obvious conclusion is that MGD provides better performance when SNR decreases, or when the background noise level increases. Regardless of the filter type, the MGD-based classification provides better performance than the k -NN based classification, when using an optimal number of filters.

Table 2. MGD and k -NN classification results for different number and compositions of filters at two different SNR values

SNR		N filters	3	6	12	18	24	30	36
Mel	7 dB	MGD	78.56	80.27	81.54	63.70	75.48	74.23	55.44
		k -NN	79.78	81.15	83.72	82.02	79.35	76.26	72.96
	2 dB	MGD	61.26	63.57	64.32	53.26	60.36	60.92	46.04
		k -NN	55.26	57.68	59.08	58.05	57.73	57.11	49.60
Linear	7 dB	MGD	84.73	97.34	98.92	82.88	94.47	83.02	59.01
		k -NN	91.45	95.79	97.46	78.78	88.04	83.00	84.35
	2 dB	MGD	74.06	83.06	79.74	63.52	76.52	65.08	49.00
		k -NN	65.89	68.76	71.03	55.26	59.71	50.77	55.49
Logarithmic	7 dB	MGD	78.74	82.36	83.04	68.02	47.24	42.04	51.77
		k -NN	70.55	75.03	75.74	73.24	57.88	54.36	42.98
	2 dB	MGD	50.21	53.88	54.98	40.70	38.56	36.87	39.81
		k -NN	48.65	52.47	54.03	39.55	36.59	35.47	37.69

Fig. 3 summarizes the results for the optimal number of filters (12 filters) as given in Table 2. Classification performance of different filter compositions for the case of using a *k*-NN or MGD classifier is clearly evident from Fig. 3. The best recognition rates are achieved by using 12 linear filters with an MGD classifier. Its advantage was especially clear when SNR was reduced to only 2 dB. The reason relates to the standard deviation of individual feature vector components. The use of 12 filters gives the best decomposition of filtered magnitude spectrum energies with the lowest possible standard deviation of individual decomposed component. Standard deviations become higher if the number of filters is lower or higher than 12. Narrow standard deviation of individual vector components provides better results.

As the results in Table 2 and Fig. 3 show the average of classification performance, the results in Tables 3 and 4 show a matrix of the classification rates. Tested objects are labeled in the first column, and the classification results are given in rows. Results of classification with 12 linear filters used in combination with MGD classifier are presented in Table 3. One misclassification occurred when using this combination at SNR of 2 dB, when a jigsaw was classified as a compressor.

The *k*-NN algorithm provides similar results, also with one misclassification for SNR of 2 dB. FCCs of vacuum cleaner 1 were misclassified as a compressor, but the other test objects were classified correctly.

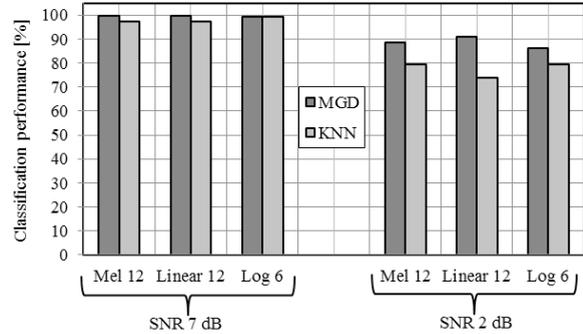


Fig. 3. Impact of different filter compositions on classification performance of MGD and *k*-NN at SNR of 7 dB and 2 dB for machinery sounds

Although *k*-NN is a simple algorithm and gives good classification results, it suffers from a few disadvantages in real-time applications. The *k*-NN algorithm requires a large database because all training feature vectors of all classes must be stored. Database size of feature vectors increases with the length of

Table 3. Matrix of MGD classification performance at optimal FCC settings

		Compressor	Vacuum cleaner 1	Jigsaw	Drill	Vacuum cleaner 2
Linear filters MGD classifier SNR 7 dB	Compressor	100	0	0	0	0
	Vacuum cleaner 1	0	99.9	0	0.1	0
	Jigsaw	4.8	0	95	0.2	0
	Drill	0	0.1	0.2	99.7	0
	Vacuum cleaner 2	0	0	0	0	100
Linear filters MGD classifier SNR 2 dB	Compressor	100	0	0	0	0
	Vacuum cleaner 1	0	98.6	0	0.2	1.2
	Jigsaw	92.6	0.1	3.5	3.8	0
	Drill	0	3.4	0	96.6	0
	Vacuum cleaner 2	0	0	0	0	100

Table 4. Matrix of *k*-NN classification performance at optimal FCC settings

		Compressor	Vacuum cleaner 1	Jigsaw	Drill	Vacuum cleaner 2
Linear filters <i>k</i> -NN classifier SNR 7 dB	Compressor	100	0	0	0	0
	Vacuum cleaner 1	0	98.6	0	0.2	1.2
	Jigsaw	0.5	0	99.5	0	0
	Drill	0	2.4	9.81	87.79	0
	Vacuum cleaner 2	0	0	0	0	100
Linear filters <i>k</i> -NN classifier SNR 2 dB	Compressor	100	0	0	0	0
	Vacuum cleaner 1	89.29	0	0	0	10.71
	Jigsaw	28.13	0	78.87	0	0
	Drill	0	2.8	20.62	76.28	0.3
	Vacuum cleaner 2	0	0	0	0	100

the training samples, as well as with the number of classes. The second disadvantage is its speed and accuracy, which are related with the determination of k -NN. The increase of the k -neighbours results in a decrease of the self-classification accuracy, in a decreased probability of misclassification, and in an increase of the computation time.

The influence of the number k (number of nearest neighbours) on the classification performance and computation time is evident from Fig. 4. Decreasing the k seems to be a reasonable choice according to the results shown on the graph, but it should be reconsidered. Finding the optimal k is essential for achieving good results in real applications.

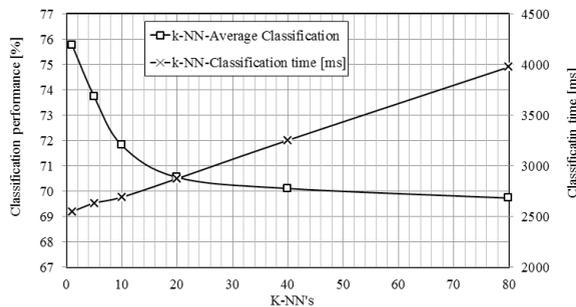


Fig. 4. Impact of k -NN on classification performance and computation time at SNR of 2 dB

Contrary to the k -NN-based classification, the MGD-based classification is faster, and its computation speed depends only on the number of reference template models (classes). The database for an individual class consists only of few statistical parameters. The small size of the database ensures a small storage size of the reference template model.

The average classification time of 2000 feature vectors using MGD is 50 ms whereas k -NN is more than 2.5 s.

The results of the classification performance of different machinery operations using an optimal number of filters are summarized in Tables 5 and 6. A comparison of the classification performances of different filter types with MGD or k -NN classifier at different SNRs is depicted in Fig. 5. Best recognition rates were achieved by using 12 mel filters with k -NN classifiers when SNR was 7 dB. Decreasing SNR to 2 dB gives better results using a linear filter type with an MGD classifier. When SNR is reduced to -2 dB, the k -NN algorithm outperforms the MGD algorithm regardless of the filter type. Mel filter type gives the best performance at SNR of 7 dB and 2 dB using the k -NN algorithm, but at -2 dB classification performance gives better results using a linear filter type.

Results in Fig. 5 clearly indicate that best overall performance is achieved if a linear or male scale filter was used.

Results in Table 5 show the classification matrix rate achieved by applying an MGD classification algorithm where different machinery operations have to be recognized. The perfect classification was achieved with an SNR of 7 dB and 2 dB. Decreasing SNR to -2 dB led to a misclassification, in which a drilling operation was classified as the operation of grinding an aluminium plate.

Table 6 shows results for the k -NN recognition of machinery operations. Very good classification performance was achieved for all SNR values with perfect classification results.

Results show that methods used in ASR systems can be successfully applied in MSR systems and can achieve high recognition accuracy even in noisy

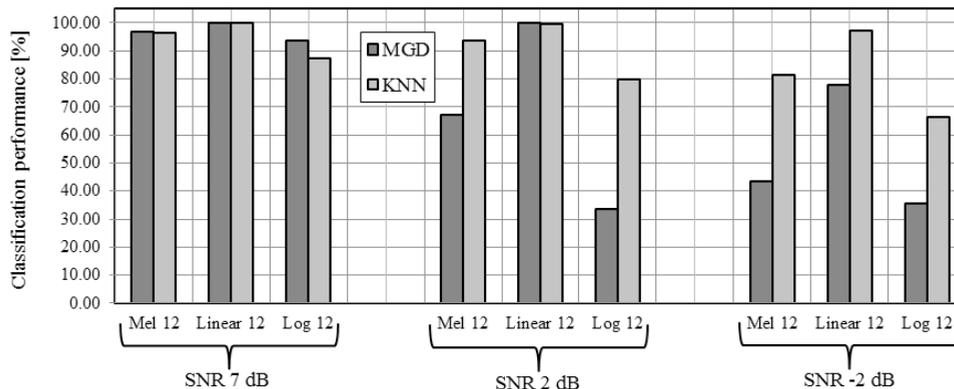


Fig. 5. Impact of different filter compositions on classification performance of MGD and k -NN at SNR of 7 dB, 2 dB and -2 dB for different operational conditions of jigsaw

conditions. A small modification of the FCC method and use of proper settings enables MSR working in real-time applications if the proper classifier is used.

Table 5. Matrix of MGD classification performance at optimal FCC settings for different jigsaw operational conditions

		Grinding	Drilling	Jigsaw - cutting
Linear-MGD classifier SNR 7 dB	Grinding	100	0	0
	Drilling	8.2	90	1.9
	Jigsaw - cutting	34.3	0.7	65
Linear-MGD classifier SNR 2 dB	Grinding	100	0	0
	Drilling	15.6	79.4	5
	Jigsaw - cutting	35.1	0.2	64.7
Linear-MGD classifier SNR -2 dB	Grinding	99.9	0	0.1
	Drilling	49.7	31.1	19.1
	Jigsaw - cutting	40	0	60

Table 6. Matrix of k -NN classification performance at optimal FCC settings for different jigsaw operational conditions

		Grinding	Drilling	Jigsaw - cutting
Linear- k -NN classifier SNR 7 dB	Grinding	81.92	16.67	1.42
	Drilling	0.11	98.26	1.63
	Jigsaw - cutting	2.61	29.25	68.14
Linear- k -NN classifier SNR 2 dB	Grinding	75	20.48	4.47
	Drilling	0	92.7	7.3
	Jigsaw - cutting	2.29	27.94	69.77
Linear- k -NN classifier SNR -2 dB	Grinding	71.46	20.59	7.95
	Drilling	0	72.55	27.45
	Jigsaw - cutting	0.65	25.65	73.69

4 CONCLUSIONS

A hypothesis proposing that automatic speech recognition algorithms can be applied for automatic machine recognition was tested. Two recognition algorithms (k -NN and MGD) were compared in combination with the FCC features extraction for three different SNRs. FCC feature extraction for automatic speech recognition based on a mel filter bank was modified through the modification of the filter bank. Two additional filter banks were developed to customize the FCC feature extraction for machinery-generated sound: linear and logarithmic. Performances of three different filter banks (mel, linear and logarithmic) were compared for the formation of

feature vectors. Individual vector components were regarded to have univariate Gaussian distribution.

Experimental results confirm the hypothesis proposing that automatic speech recognition algorithms can be used for automatic machine recognition. The highest recognition rates of different machines classification were achieved by the MGD classifier in combination with linear filter type regardless of SNR. The results in Fig. 3 clearly show that a linear filter type in combination with MGD classifier should be used for machinery identification. However, classification performance of different machinery operations was better when using the k -NN algorithm in combination with male scale filter type, when SNR was low. When decreasing SNR to -2 dB, the linear filter type proves to be a better option.

Overall results of the experiment show that machinery noise features should be extracted using the proposed linear filter bank.

Despite the good classification results of different machinery operations, the k -NN might not be the best solution to use. Because it has no statistical model, it might not work well when a few more machinery operations would be added to an MSR procedure. Taking k -NN speed and database size into account, definitely makes the MGD algorithm more appropriate.

Algorithms of the proposed procedure are very fast, and the recognition can be performed in real-time. In further work, a full Gaussian mixture model should be tested for a further upgrade of machine classification performance.

The proposed procedure was already used in the system for the automatic classification of noise events during environmental noise measurements. It was also tested for the detection of cavitation phenomena on kinetic pumps and the quality inspection of suction units used in vacuum cleaners.

5 REFERENCES

- [1] Germen, E., Basaran, M., Fidan, M. (2014). Sound based induction motor fault diagnosis using Kohonen self-organizing map. *Mechanical Systems and Signal Processing*, vol. 46, no. 1, p. 45-58, DOI:10.1016/j.ymssp.2013.12.002.
- [2] Lu, W., Jiang, W., Yuan, G., You, L. (2013). A gearbox fault diagnosis scheme based on near-field acoustic holography and spatial distribution features of sound field. *Journal of Sound and Vibration*, vol. 332, no. 10, p. 2593-2610, DOI:10.1016/j.jsv.2012.12.018.
- [3] Amarnath, M., Sugumaran, V., Kumar, H. (2013). Exploiting sound signals for fault diagnosis of bearings using decision tree. *Measurement*, vol. 46, no. 3, p. 1250-1256, DOI:10.1016/j.measurement.2012.11.011.

- [4] Benko, U., Petrovic, J., Jurcic, D., Tavcar, J., Rejec, J. (2005). An approach to fault diagnosis of vacuum cleaner motors based on sound analysis. *Mechanical Systems and Signal Processing*, vol. 19, no. 2, p. 427-445, DOI:10.1016/j.ymssp.2003.09.004.
- [5] O'Shaughnessy, D. (2008). Invited paper: Automatic speech recognition: History, methods and challenges. *Pattern Recognition*, vol. 41, p. 2965-2979, DOI:10.1016/j.patcog.2008.05.008.
- [6] Furui, S. (2005). 50 years of progress in speech and speaker recognition research. *ECTI Transactions on Computer and Information Technology*, vol. 1, no. 2, DOI:10.1121/1.4784967.
- [7] Goldhor, R.S. (1993). Recognition of Environmental Sounds. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, p. 149-152, DOI:10.1109/ICASSP.1993.319077.
- [8] Chu, S., Narayanan, S., Kuo, J. (2009). Environmental sound recognition with time-frequency audio features. *IEEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 6, DOI:10.1109/TASL.2009.2017438.
- [9] Chachada, S., Kou, J. (2013). Environmental Sound Recognition: A Survey. *Signal and Information Processing Association Annual Summit and Conference*, DOI:10.1109/APSIPA.2013.6694338.
- [10] Ntalampiras, S., Potamitis, I., Fakotakis, N. (2008). *New Directions in Intelligent Interactive Multimedia*. Springer, Berlin, Heidelberg, p. 147-153, DOI:10.1007/978-3-540-68127-5_15.
- [11] Boela, Y., Grau, A., Pelissier, A., Sanfeliu, A. (2004). *Progress in Pattern Recognition, Image Analysis and Applications*. Springer, Berlin, Heidelberg, p. 287-295, DOI:10.1007/b101756.
- [12] Kankar, P.K., Sharama, S.C. Harsha, S.P. (2011). Rolling element bearing fault diagnosis using wavelet transform. *Neurocomputing*, vol. 74, p. 1638-1645, DOI:10.1016/j.neucom.2011.01.021.
- [13] Kakar, V.K., Kandpal, M. (2013). Techniques of acoustic feature extraction for detection and classification of ground vehicles. *International Journal of Emerging Technology and Advance Engineering*, vol. 3, no. 2, p. 419-426.
- [14] Balsamo, L., Betti, R., Beigi, H. (2014). A structural health monitoring strategy using cepstral features. *Journal of Sound and Vibration*, vol. 333, p. 4526-4542, DOI:10.1016/j.jsv.2014.04.062.
- [15] Marquez-Molina, M., Sanchez-Fernandez, L.P., Suarez-Guerra, S., Sanchez-Perez, L.A. (2014). Aircraft take-off noises classification based on human auditory's matched features extraction. *Applied Acoustic*, vol. 84, p. 83-90, DOI:10.1016/j.apacoust.2013.12.003.
- [16] Prezelj, J. (2015). Patent SI 24518 A. *Sistem za avtomatski monitoring hrupa in za klasifikacijo virov hrupa v opazovanem okolju (Eng: System for Automatic Noise Source Identification and Classification)*. Slovenian Intellectual Property Office, Ljubljana.
- [17] Čudina, M., Prezelj, J. (2009). Detection of cavitation in situ operation of kinetic pumps: effect of cavitation on the characteristic discrete frequency component. *Applied Acoustics*, vol. 70, no. 9, p. 1175-1182, DOI:10.1016/j.apacoust.2009.04.001.
- [18] Peeters, G., Rodet, X. (2002). Automatically selecting signal descriptors for sound classification. *International Computer Music Conference*, Goteborg.
- [19] Islam, R., Rahman, F. (2009). Codebook design method for noise robust speaker identification based on genetic algorithm. *International Journal of Computer Science and Information Security*, vol. 4, no. 1.
- [20] Ganchev, T. (2011). *Contemporary Methods for Speech Parameterization*. Springer, New York, Dordrecht, Heidelberg, London, DOI:10.1007/978-1-4419-8447-0.
- [21] Zheng, F., Zhang, G., Song, Z. (2001). Comparison of Different Implementation of MFCC. *Journal of Science & Technology*, vol. 16, no. 6, p. 582-589, DOI:10.1007/BF02943243.
- [22] Sigurdsson, S., Petersen, K.B., Lehn-Schioler, T. (2006). Mel frequency coefficients: An evaluation of robustness of MP3 encoded music. *Proceedings of the 7th International Conference on Music Information Retrieval*.
- [23] Muda, L., Begam, M. Elamvazuthi, I. (2010). Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques. *Journal of Computing*, vol. 2, no. 3, p. 138-143.
- [24] Mengusoglu, E. (2015). Speaker model adaptation based on confidence score. *Tehnički vjestnik - Technical Gazzete*, vol. 22, no. 4, p. 873-878, DOI:10.17559/TV-20140120095957.
- [25] Saini, I., Singh, D., Khosla, A. (2013). QRS detection using k-nearest neighbor algorithm (KNN) and evaluation on standard ECG database. *Journal of Advance Research*, vol. 4, p. 331-334, DOI:10.1016/j.jare.2012.05.007.
- [26] Izenman, A.J. (2008). *Modern Multivariate Statistical Techniques*. Springer Science+Buisness Media, New York, DOI:10.1007/978-0-387-78189-1.