

Single Sensor Acoustic Feature Extraction for Embedded Realtime Vehicle Classification

Andreas Starzacher and Bernhard Rinner
Institute of Networked and Embedded Systems
Klagenfurt University, Austria
 {andreas.starzacher, bernhard.rinner}@uni-klu.ac.at

Abstract—Vehicle classification is an important task for various traffic monitoring applications. This paper investigates the capabilities of acoustic feature generation for vehicle classification. Six temporal and spectral features are extracted from the audio recordings and six different classification algorithms are compared using the extracted features. We focus on a single sensor setting to keep the computational effort low and evaluate its classification accuracy and realtime performance. The experimental evaluation is performed on our embedded platform using recorded data of about 150 vehicles. The results are applied in our ongoing research on fusing video, laser and acoustic data for realtime traffic monitoring.

Keywords-acoustic feature extraction; signal processing; vehicle classification; embedded sensor fusion;

I. INTRODUCTION

Multisensor data fusion (MSDF) techniques are found in various applications to combine homogeneous and heterogeneous sensor information. The main objective of MSDF is to exploit multiple source sensor data to increase the robustness and confidence as well as extend the spatial and temporal coverage [1], [2]. In time-critical applications, such as traffic monitoring, it is essential to achieve robust and accurate results within limited periods of time. There is a huge pool of various fusion algorithms ranging from estimation methods to different types of classification and inference techniques. However, due to resource requirements not all of them are applicable to embedded realtime fusion.

In our ongoing research project *Autonomous Traffic Monitoring by Embedded Vision (EVis)* we apply MSDF techniques to fuse heterogeneous data originating from various sources such as audio, visual and laser sensors. MSDF is performed on a network of embedded devices with the ultimate objective to improve vehicle classification [3].

In this paper, however, we focus on a single acoustic sensor setting in order to investigate the capabilities of acoustic feature generation for vehicle classification (similar setup found in [4]). We evaluate its classification accuracy and realtime performance. In more detail, we present six temporal and spectral acoustic features that have discriminative characteristics concerning vehicle classes such as cars and trucks. This feature vector serves then as input for six different classifiers. The feature extraction and classification

algorithms are evaluated using real world test data of about 150 vehicles from highway traffic. All processing is executed on an embedded platform, and the realtime deadline for the complete classification is given as 1000 ms. Our approach achieves classification rates between 93 % and 97 % within this deadline. However, our approach has some limitations in multi-lane separation.

There are several examples for related work. In [5] data fusion of acoustic and visual data is applied to vehicle tracking. Different to our single acoustic sensor setup, they use an audio beamforming array of four microphones to determine the direction and location of vehicles. In [6] the classification of moving vehicles using acoustic signatures and data fusion is investigated. Acoustic vehicle classification is investigated as well in [7]. They apply a Gaussian mixture model which is motivated by the ability of modeling arbitrary distributions for varying operation modes and engines of motor vehicles. [8] uses two sets of features. The first set consists of several harmonic components characterizing the noise of the engine and the second set consists of key frequency components for characterizing such as the friction noise of tires.

The remainder of the paper is structured as follows: Section II describes the acoustic feature extraction algorithms in detail. Section III gives a rough description of several feature fusion algorithms that are used for evaluation purpose to show the discriminative characteristics of the acoustic features. Experimental studies, evaluation and description of the data and study location are presented in Section IV. The paper concludes with a brief summary.

II. FEATURE EXTRACTION

A typical classification system consists of the following components: sensor data acquisition, preprocessing, feature extraction, classification and postprocessing.

The data acquisition task is done with our embedded multisensor data fusion platform interfacing acoustic, visual and laser sensors. For a detailed description of our embedded platform, the structure of our sensor network and the three-layered hierarchical fusion architecture see [9].

Next we have the preprocessing task which in our case downsamples the input audio sample (more details in Section

IV-A). After preprocessing features are extracted. If the feature extraction results in highly class-discriminative features, it can make the classification task trivial. But, however, dealing especially with acoustic data it is to point out that it is very difficult to end up with robust acoustic features, because of the non-stationarity of the acoustic signals and the inference due to environmental noise (e.g., vehicles on neighboring lanes on a freeway). Hence, the quality of features in terms of robustness is highly domain-specific.

We implemented and evaluated several acoustic features in different representational domains, i.e., time, spectral and cepstral domain. Time domain features are zero-crossing rate and short time energy (*ste*). Spectral features are spectral centroid, spectral bandwidth (*spbw*), spectral roll-off point (*spro*), spectral flux and band energy ratio values (*ber*). Additionally, a cepstral analysis (*cep*) is performed as well. Based on previous evaluation results [10], considering the degree of class-separability of each individual feature, we take *ste*, *spbw*, *spro* and two values of *ber* coefficients and the *cep* coefficient for our feature fusion task. These features additionally allow for embedded processing as well (see Section IV-B).

In the following we list and explain the different features from time and frequency domain that we use in the complete classification process.

A. Time domain feature

As mentioned above, the *short time energy* value is an acoustic feature in time domain. It is discriminative between different types of vehicles and not computationally expensive. The definition of this feature is given in eq. 1. Single energy values represent local information (short time characteristics). In order to extract features over longer time scales statistical moments are to be computed. Since this feature is sensitive to noise because of taking the amplitude of the signal into consideration, we further use spectral and cepstral features for the classification task.

$$E_t = \frac{1}{N} \sum_{n=0}^{N-1} |x_t(n)|^2, \quad (1)$$

where E_t is the energy of the t^{th} input block, N the total number of samples in a block and $x_t(n)$ the t^{th} input energy value of the audio sample.

B. Frequency domain features

Vehicles emit acoustic sounds at various frequencies. Therefore the spectrum of a signal that is generated typically by applying a Fourier transform is a powerful feature. In our feature extraction process we extract the spectral bandwidth, spectral roll-off and band energy ratio values.

The *spectral bandwidth* value measures the spread of frequencies around the spectral centroid. We use the following

definition (eq. 2) for implementing the feature extraction algorithm on our embedded platform:

$$spbw = \sqrt{\frac{\sum_{n=0}^{N-1} (n - spct_t)^2 \cdot |X_t(n)|^2}{\sum_{n=0}^{N-1} |X_t(n)|^2}}, \quad (2)$$

where *spct* is the spectral centroid of the t^{th} input block and $X_t(n)$ the FFT of the t^{th} input signal frame.

The *spectral roll-off* point is an additional feature in frequency domain and is defined by eq. 3. It is a measure indicating up to what frequency α percent of the spectrum is summed up, i.e., a higher roll-off value corresponds with more intense or higher frequencies. In other words, the sum of the spectrum up to the roll-off frequency approximates α percent of the total spectrum.

$$\sum_{n=0}^{R_t-1} |X_t(n)|^2 = \frac{\alpha}{100} \sum_{n=0}^{N-1} |X_t(n)|^2, \quad (3)$$

where α is typically 85 (as in [11]) and $R_t - 1$ represents the roll-off value.

The last feature in frequency domain we use is the band energy ratio values (eq. 4) which describe the ratio of energy in certain frequency bands to the total signal energy.

$$ber_{t,i} = \frac{\sum_{n \in S_i} |X_t(n)|^2}{\sum_{n=0}^{N-1} |X_t(n)|^2}, \quad (4)$$

where $ber_{t,i}$ is the band energy ratio of the i^{th} subband and S_i the set of Fourier transform coefficients that belong to the i^{th} subband. We only used the 6th and 7th subband, i.e., $S_6 = [N/4, N/2]$ and $S_7 = [N/2, N]$.

In addition to time and spectral features we performed a cepstral analysis as well. This is a convenient way to model spectral energy distributions. The definition of a cepstrum of a signal is defined as (see eq. 5)

$$cep(t) = F^{-1}\{\log(|F\{x_t\}|)\}, \quad (5)$$

where F^{-1} is the inverse of the Fourier transform.

III. FEATURE FUSION

After extracting discriminative acoustic features to describe the characteristics of the two classes of vehicles (cars and trucks), we pool these features into a single feature vector \mathbf{x} consisting of six features in total, i.e., *ste*, *spbw*, *spro*, *cep* and two coefficients of *ber*. This vector \mathbf{x} is used for the classification task. In our experiments we have implemented several well-known feature fusion algorithms to perform the classification task. These algorithms are feedforward neural network, support vector machine, naive Bayes approach, linear and quadratic discriminant analysis and k-nearest neighbor.

A. Feedforward Neural Network

The general structure of the artificial neural network (ANN) is shown in Fig. 1.

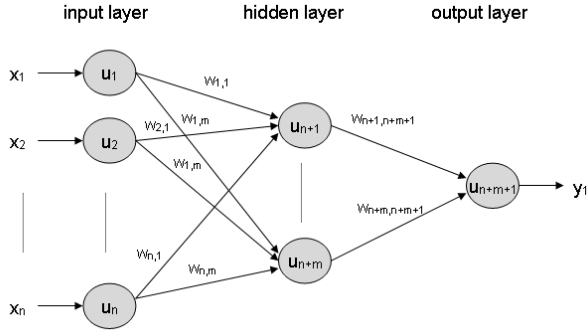


Figure 1. A feedforward neural network

Most of the parameters of the ANN are determined based on several heuristics, e.g., choosing the number of hidden layers and neurons. Our ANN consists of an input, a hidden and an output layer. Three layers are sufficient to approximate any function in an arbitrary accuracy. The number of input neurons is determined by the number of features that originate from a single sensor node. There is a single output neuron which represents the decision (car, truck). The number of hidden neurons is specified by the following heuristics: $H = \lfloor (N + M) \cdot d \rfloor$ and $H = \lfloor (N + M) \cdot d \rfloor + 1$, where N is the number of input neurons, M the number of output neurons, d a constant with $d \in (0, 1]$. As we are interested in embedded realtime processing (fusion), the training of the ANN is performed offline. Training a ANN can be computationally expensive as well as time consuming. The training time spent depends on several parameters like number of layers, neurons and epoches, the type of activation functions and the training algorithm itself.

For our case studies in Section IV-B the parameters of the ANN are set as follows: $N = 6$, $M = 1$, H as given by heuristics with $d = 2/3$. Hence, the input feature vector to the ANN is $\mathbf{x} = (x_1, x_2, x_3, x_4, x_5, x_6)^T$, where $x_1 = ste$, $x_2 = spbw$, $x_3 = spro$, $x_4 = cep$ and x_5, x_6 are the two band energy ratio coefficients (*ber*). Additionally, fixed parameters are the number of training epoches which is set to 10^5 and the activation function set to a Gaussian.

B. Support Vector Machine

In addition to a ANN, we implement a support vector machine (SVM). In literature SVMs are widely used in classification systems and applied to data fusion applications as well [10]. Basically, SVM tries to obtain an optimal (maximum) margin, i.e., a maximum margin width, between the classes of interest, hence resulting into a maximum separating (hyper)plane.

Especially in our case studies we use the following parameters for the SVM: cost parameter = 1, tolerance of termination criterion = 10^{-3} and the kernel type is either a radial basis or linear function. Hence, the degree in kernel function can be neglected as we do not apply a polynomial kernel.

C. Naive Bayes Approach

The naive Bayes approach has several advantages like its simplicity and transparency. In contrast to Bayesian networks (a naive Bayes classifier is the simplest form of a Bayesian network [12]) with naive Bayes classification conditional independence of the feature values x_i of \mathbf{x} is assumed. Accordingly, the probability of \mathbf{x} given a certain group g_i is calculated by multiplying the probabilities of each x_i . It is important to see that the posterior probability is proportional to a certain group prior p_i multiplied by the product of the appropriate (independent) likelihoods conditioned on g_i . We choose to implement a Gaussian for each group with estimated parameters ($\hat{\mu}$, $\hat{\sigma}^2$) from the training set. Finally, we apply a maximum-a-posteriori (*map*) rule to obtain the final decision $\hat{D}(\mathbf{x})$, i.e., the group label (see eq. 6).

$$\hat{D}(\mathbf{x}) = \arg \max_i p(g_i | \mathbf{x}) \quad (6)$$

In case the product of likelihoods is very small or converge to zero, we compute the product with logarithms (log posteriors). Even though in many applications the conditional independence is not given, classification based on naive Bayes gives reasonable results [13] (see Section IV-B).

D. Discriminant Analysis

The classical approach of linear discriminant analysis (LDA) is to find an optimal projection of features by minimizing the within-class and maximizing the between-class distance. Quadratic discriminant analysis (QDA) is quite similar to LDA, but it allows for quadratic decision boundaries between classes. Thus, the classification rule is quite intuitive, namely classify \mathbf{x} in that class under which the discriminant function gives the largest value. The estimated discriminant functions $\hat{d}_g(\mathbf{x})$ for LDA and QDA are explained in [9] in detail. We only show the classification rule as given in eq. 7. It is the same for both LDA and QDA, respectively. The class label $\hat{D}(\mathbf{x})$ equals g^* iff g makes the discriminant function $\hat{d}_k(\mathbf{x})$ a maximum.

$$\hat{D}(\mathbf{x}) = g^* : \Leftrightarrow g^* = \arg \max_g \hat{d}_g(\mathbf{x}) \quad (7)$$

E. k-Nearest Neighbor

Due to the plausible fact that for any number of classes (groups) it may be generalized that half of the total information needed for the classification task is contained in the nearest neighbor(s) [14], we implemented a k-Nearest Neighbor classifier (KNN) with k chosen as 20% of the training data. Although we have already shown that KNN is not applicable to embedded realtime feature fusion [9], we want to show that from a classification error rate point of view KNN keeps reasonably pace with more sophisticated feature fusion algorithms like those mentioned above. For future work, in order to keep the training time small and expand KNN for realtime classification, a Voronoi-based KNN approach might be implemented [15].

A detailed experimental evaluation of the algorithms described above on other datasets for embedded realtime feature-based fusion on our embedded platform can be found in our previous work [9], [3].

IV. EXPERIMENTAL STUDIES AND EVALUATION

A. Data and Study Location

The acoustic feature extraction and feature fusion algorithms as described in Section II and III are applied to real-world datasets that have been recorded on a freeway bridgeover. A rough draft of the experimental setup is depicted in Fig. 2. The freeway has two lanes in both directions. A directional microphone was placed on the bridgeover (mounted on the embedded prototype platform represented by the black circle) and directed to the center of the outer lane (approximately 10 meters to the highway lane). The classification process is done for this outer lane exclusively. The intention was to record acoustic data from cars and trucks under environmental noise (due to vehicles on neighboring lanes in the same and opposite direction). The audio recording was performed with a sampling rate of 44,1 kHz in mono format with 16 bit resolution. The final audio input of the feature extraction algorithms is defined as follows: sampling rate: 8 kHz (downsampling performed in the preprocessing task, see Section II), resolution: 16 bit, channels: *mono* and the maximum audio sample duration is limited to 4 seconds. The vehicles of interest are cars and trucks.

B. Results

In Table I the individual extraction time statistics (mean and standard deviation) for both cars and trucks are summarized. $\mu_{e,car}$ and $\sigma_{e,car}$ represent the mean extraction time and the standard deviation for cars ($\mu_{e,truck}, \sigma_{e,truck}$, analogously for trucks). The high standard deviation can be observed due to the varying duration of the different audio samples (as already mentioned, with a maximum duration set to 4 seconds). Trucks are typically driving slower than cars and hence are located in the recording region for a longer period of time.

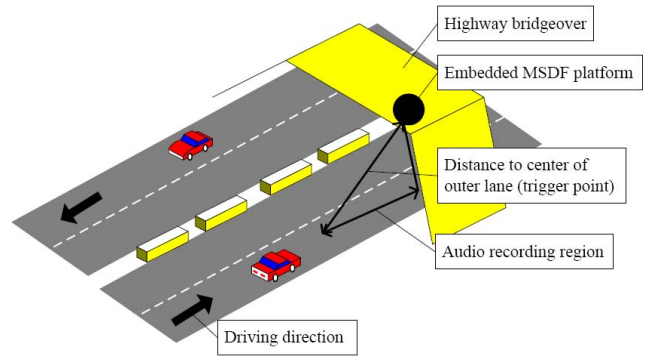


Figure 2. Experimental Setup

Therefore, the mean extraction time of cars and trucks differ since the recording time of trucks (duration of audio samples of trucks) is longer than it is with cars.

Table I
INDIVIDUAL EXTRACTION TIME STATISTICS [IN MS]

Extraction Algorithm	$\mu_{e,car}$	$\sigma_{e,car}$	$\mu_{e,truck}$	$\sigma_{e,truck}$
<i>ste</i>	14.62	2.4	18.87	3.74
<i>ber</i>	149.09	25.38	189.82	37.3
<i>spbw</i>	149.76	25.36	192.08	38.01
<i>spro</i>	151.57	25.64	193.18	38.16
<i>cep</i>	281.6	47.86	362.17	74.87

As we are interested in the total time spent for the whole classification process including feature extraction, we separately evaluated the time that is needed to classify a feature vector consisting of the six acoustic features presented in Section II. The results can be seen in Table II, where μ_c is the mean classification time and σ_c the standard deviation. SVM and ANN slightly outperforms QDA, LDA and NBC. KNN is computationally more expensive approximately by a multiplying factor of 7.4. As shown in [9] KNN is not suitable for embedded realtime fusion if the number of training data and the size of the feature vector is large. But, however, in the case of a feature vector of size = 6 and a relatively small training set (smaller than 1000 samples), KNN does meet realtime constraints very well as the values in Table II indicate.

Comparing extraction and classification time it is worth mentioning that the major percentage of the total time of the feature-based fusion process is used for feature extraction. The percentage of the classification task does not exceed 0.07%, except KNN with an upper bound of 0.4%. Hence, the classification task for a feature vector of six entries is of no consequence regarding the complete processing time. Table III summarizes the time used for feature extraction and feature-based fusion for each individual classification algorithm. The second column represents mean processing

Table II
INDIVIDUAL CLASSIFICATION TIME STATISTICS [IN MS]

Classification Algorithm	μ_c	σ_c
SVM	0.3089	0.0425
ANN	0.3094	0.0376
QDA	0.4302	0.0528
LDA	0.4318	0.0540
NBC	0.4613	0.0485
KNN	2.9598	0.1175

time for cars and the third column the one for trucks (all times given in ms).

Table III
TOTAL EXTRACTION AND CLASSIFICATION TIME STATISTICS [IN MS]

Classification Algorithm	$\mu_{e,car} + \mu_c$	$\mu_{e,truck} + \mu_c$
SVM	746.95	956.43
ANN	747.95	957.43
QDA	749.07	958.55
LDA	750.07	960.58
NBC	751.10	960.58
KNN	754.59	964.07

A further objective of the experimental study is to evaluate and discuss the combination of the six acoustic features concerning the discriminability between cars and trucks. As mentioned before this is done in terms of classification error rates and confusion matrices for the individual algorithms. The classification error rates represent the proportion of false positives and false negatives to the total number of samples of particular test sets. In order to derive meaningful classification error rates we perform a 5-times cross-validation. The results of the cross-validation are depicted in Fig. 3.

We achieved a notable classification rate of 96.6% with the artificial neural network (ANN). Furthermore, even though the conditional independence assumption does not hold for our acoustic features, the naive Bayes classifier (NBC) works very well. With a classification rate of 95.9% (LDA with 95.2% as well) it outperforms even more sophisticated algorithms like SVM (94.5%). Moreover, the k-nearest neighbor approach is also able to compete with these algorithms. Even though it constructs a very intuitive and simple separation between classes, it reaches a classification rate up to 93.1%. QDA is not satisfyingly applicable to our acoustic data (under 90%).

C. Limitations

Although the classification based on acoustic features is very promising, there are some limitations on the use of single audio sources. The first challenge in using single audio source is depicted in Fig. 4(a) and (b). In our case the classification is performed exclusively for vehicles driving on the outer lane. If the vehicle is driving (a) at the inner lane

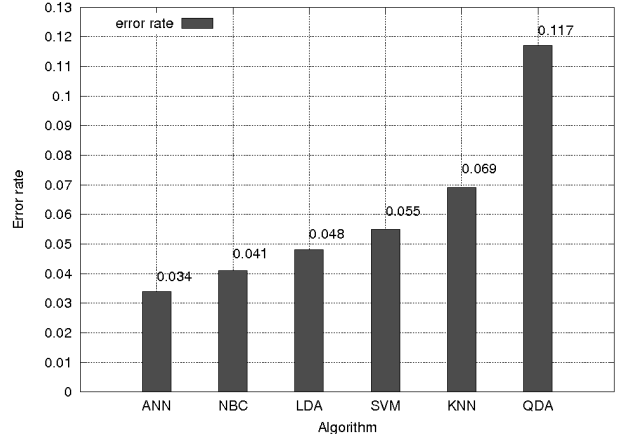


Figure 3. Classification error rates (true positives + true negatives)

Table IV
CONFUSION MATRICES

ANN	19	1	LDA	19	2
	1	9		1	8
SVM	19	3	QDA	18	1
	1	7		2	9
LDA	19	3	KNN	18	2
	1	7		2	8

or (b) two vehicles are driving in parallel, we cannot reliably determine if the signal originates from vehicles driving on the inner or outer lane. Therefore we use an additional laser sensor that functions as a trigger for the audio recordings for a specific lane and as a plausibility check that a vehicle is driving on that lane.

If cars and trucks are driving in parallel (as shown in Fig. 4(c)), it is very hard to classify cars as cars due to the fact that trucks are typically producing dominantly much louder sounds than cars. To the contrary, if cars and trucks change lanes, there are no problems to classify trucks correctly, because the sound of the car is of insignificant consequence concerning inferences.

Additionally, there are also limitations concerning the computational performance of the algorithms and the underlying embedded platform (physical limitations). As presented in Section IV-B the maximum execution time for extraction and classification for cars is ranging from 747ms to 755ms and for trucks from 957ms to 965ms (see Table III). If vehicles are driving too closely in succession exceeding the physical time spent to extract and classify one particular vehicle, it is physically not possible to acquire data from all successive vehicles (Fig. 4(d)).

Currently we perform the complete extraction and classification tasks after a particular recording time duration. This obviously affects the realtime behavior of the system. An improvement would be to combine the capturing and

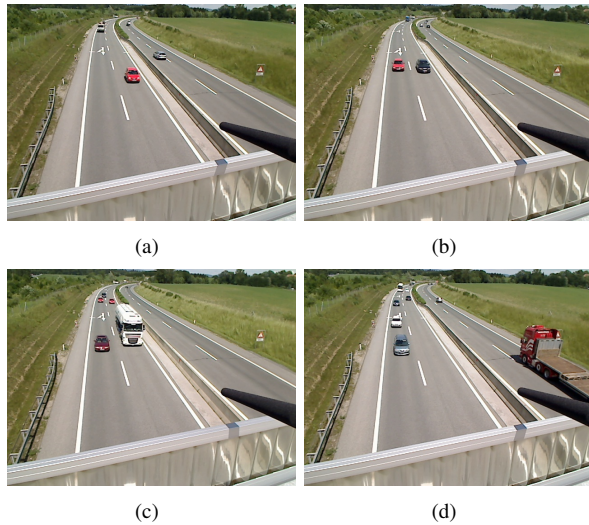


Figure 4. Problematic test cases for lane accurate vehicle classification using single acoustic sensor

processing tasks performing the extraction block by block in parallel with the audio recordings.

V. CONCLUSION

As part of our ongoing work on embedded realtime fusion of heterogeneous sensor data for vehicle classification applied to traffic monitoring, this paper investigates the capabilities of acoustic feature generation for vehicle classification. A total of six temporal and spectral features are extracted from the audio recordings. The extraction algorithms are described in detailed. Standard classification algorithms are compared using these features. Based on previous work the classification algorithms chosen are a neural network, a support vector machine, a naive Bayes classifier, a k-nearest neighbor approach and a linear as well as quadratic discriminant analysis.

We focus on evaluating the classification accuracy (classification rates and confusion matrices) as well as realtime performance (extraction and classification execution time) of the single sensor setting approach on our embedded platform. The results concerning both classification accuracy and realtime performance of the algorithms are very promising. Due to these results we apply them in our ongoing research on fusing acoustic with visual and laser data.

ACKNOWLEDGMENT

This research is supported by a grant from the Austrian Research Promotion Agency under the FIT-IT[visual computing] grant 813399.

REFERENCES

[1] H. Ruser and F. P. León, "Informationsfusion - Eine Übersicht," *tm - Technische Messen, Oldenburg*, vol. 74, no. 3, pp. 093–102, 2007.

[2] H. Hu and J. Q. Gan, "Sensors and Data Fusion Algorithms in Mobile Robotics," *Technical Report: CSM-422, Department of Computer Science, University of Essex, Colchester CO4 3SQ, United Kingdom*, 2005.

[3] A. Starzacher and B. Rinner, "Embedded Realtime Feature Fusion based on ANN, SVM and NBC," in *Proceedings of the Twelfth International Conference on Information Fusion (Fusion-2009)*, Seattle, WA, USA, 2009.

[4] A. Y. Nooralahiyan, H. R. Kirby, and D. McKeown, "Vehicle Classification by Acoustic Signature," *Mathematical and Computer Modelling*, vol. 27, no. 9, pp. 205–214, 1998.

[5] M. Kushwaha, O. Songhwai, I. Amundson, X. Koutsoukos, and A. Lediczi, "Target Tracking in Heterogeneous Sensor Networks Using Audio and Video Sensor Fusion," in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Seoul, 2008, pp. 14–19.

[6] M. F. Duarte and Y. H. Hu, "Vehicle classification in distributed sensor networks," *Journal of Parallel and Distributed Computing*, vol. 64, no. 7, pp. 826–838, 2004.

[7] B. F. Necioglu, C. T. Christou, B. E. George, and G. M. Jacyna, "Vehicle Acoustic Classification in Netted Sensor Systems Using Gaussian Mixture Models," in *Proceedings of SPIE, Conference No.14, Signal processing, sensor fusion, and target recognition*, Orlando, Florida, USA, 2005, pp. 409–419.

[8] G. Baofeng, M. Nixon, and T. Damarla, "Acoustic Information Fusion for Ground Vehicle Classification," in *11th International Conference on Information Fusion*, Cologne, Germany, 2008, pp. 1–7.

[9] A. Starzacher and B. Rinner, "Evaluating KNN, LDA and QDA Classification for embedded online Feature Fusion," in *Proceedings of the Fourth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP-2008)*, Sydney, NSW, 2008, pp. 85–90.

[10] A. Klausner, A. Tengg, and B. Rinner, "Distributed multilevel data fusion for networked embedded systems," *IEEE Journal on Selected Topics in Signal Processing*, vol. 2, no. 4, pp. 538–555, 2008.

[11] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," in *IEEE Transaction on Speech and Audio Processing*, 2002, pp. 293–302.

[12] D. Geiger, M. Goldszmidt, G. Provan, P. Langley, and P. Smyth, "Bayesian Network Classifiers," in *Machine Learning*, 1997, pp. 131–163.

[13] J.-N. Sulzmann, J. Fürnkranz, and E. Hüllermeier, "On Pairwise Naive Bayes Classifiers," in *Proceedings of the 18th European Conference on Machine Learning*, Warsaw, Poland, 2007, pp. 371–381.

[14] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 021–027, 1967.

[15] M. Kolahdouzan and C. Shahabi, "Voronoi-based k nearest neighbor search for spatial network databases," in *VLDB-2004*, vol. 30, Toronto, Canada, 2004, pp. 840–851.