Neuro-Fuzzy based Techniques for Acoustic Signal Classification

V.S. KODOGIANNIS
Department of Computer Science, University of Westminster
Harrow, Middlesex, HA1 3TP
UNITED KINGDOM

Abstract: Neural networks and fuzzy systems are currently finding practical applications, ranging from ‘soft’ regulatory control in consumer products to accurate modelling of non-linear systems. This paper presents the design of a classification system for vehicle acoustic signal classification. Vehicle acoustic signals have long been considered as unwanted traffic noise. Acoustic signals generated by each vehicle have been used to detect its presence and classify its type. Training and testing data used in this paper were collected from roadside sensor station at the Valle d’Aosta highway in north-western Italy. Two main systems, multilayer perceptron networks and adaptive fuzzy logic systems, are considered, analysed, and used as classifiers. The results indicate that the fuzzy classifier based on a novel proposed defuzzification method, namely balance of area (BOA), provide more accurate classifications compared to the conventional classifier systems.

Keywords: Acoustic classification, adaptive fuzzy systems, defuzzification methods.

1. Introduction
The overall acoustic signal of a vehicle arises from several sources, which include the engine, gears, fan, cooling system, road-tire interaction, exhaust, brakes, and aerodynamics effects [1]. However of great importance is not only the classification of acoustic signals, but also the source that generates the acoustic signals, such as vehicle classification and speaker verification. This information can be used to forecast pavement maintenance needs. Besides road surface maintenance, this vehicle classification information is also used to estimate in cost responsibility studies to allocate the costs of building and maintaining the highway systems across classes of vehicles in proportion to their responsibility for these costs, e.g. developing tax rates for motor vehicles. Other uses of vehicle classification data are to compare the accident rates of different classes of vehicles and to analyse road capacity, determining the width and number of lanes required and the need for additional climbing lanes, and designing intersections and interchanges.

The most active research on classification of acoustic signals using artificial neural networks and fuzzy logic systems is in the fields of the speech recognition and underwater acoustics [2-3]. Most neural networks and fuzzy logic systems applications in the speech recognition field are subtasks such as phoneme classification, voice-unvoice-silence discrimination. On the other hand, sonar applications are generally separated according to whether the sensing mode is active or passive. More recent approaches to sonar signal processing and target classification have incorporated non-linear techniques and in the case of sonar imaging have adopted fuzzy imaging techniques [4].

Fuzzy logic techniques can be used in several stages in pattern classification problems, and they are considered as another supporting tool in solving pattern classification problems. The fuzzy logic systems usually have problems with defuzzification part. Various techniques have been investigated, like centroid of area (COA), centre average (CA), modified centre average (MCA) [5].

The goal of this study is to build classifiers, which use a training algorithm to adapt its parameters in order to solve the vehicle classification problem using acoustic signals. This research will utilise adaptive fuzzy logic systems and neural networks as classifiers. The objective of this study is to develop new techniques of fuzzy neural classifiers, to investigate and extract significant features from acoustic signals and ultimately to design a robust, effective, and reliable vehicle classification system. A new defuzzification method will be developed for the proposed MIMO adaptive fuzzy logic systems. This technique is named balance of area (BOA) defuzzification. This defuzzification method can be used either with standard fuzzy logic system with fuzzifier and defuzzifier or can be used with MIMO AFLS.

2. Feature Extraction and Analysis

2.1 Acoustic Sensor Design and Data Acquisition
An acoustic classification system design composes of the acoustic sensor design and data acquisition, feature analysis and extraction, classifier design, and implementation. The vehicle acoustic classification
system consists of several sub-systems as shown in Fig. 1. The definition of significant features plays an important role in the classification problem.

![Diagram of Vehicle Acoustic Classification System]

Feature extraction depends on the problem at hand; there is no universal rule for feature extraction. At this point, feature analysis and extraction still rely on humans to analyse and extract good features from the process. In the vehicle classification problem, vehicles must be detected first before they can be classified. The detection process is designed to detect the presence of a passing vehicle and to initialise and end the feature extraction process. Since different classes of vehicles have different sizes, the ending point of feature extraction was adapted according to an initial prediction by a fuzzy logic system. The information of duration and loudness are used as the inputs to this prediction system. After a vehicle is detected significant features are extracted. All features are time domain features extracted directly from acoustic energy information.

The vehicle acoustic signals have a wide range in frequencies as shown in Fig. 2. Most of the acoustic energy is in low frequencies. Incoming signals are processed one window at a time.

![Magnitude Response of an Acoustic Signal]

A rectangular window is used in this case. The size of the window is 0.02 seconds. Since the sampling rate is 44100 Hz, therefore, there are 882 samples in one window.

The signal in this window is filtered by a high-pass filter having a cut-off frequency of 2700 Hz: a type I sixth-order Chebyshev filter with 0.1dB ripple. There are two filters because we want to suppress the signals in lower frequencies more than the higher frequencies. Using these filters, the circular array ideally forms a beam pattern corresponding to the signals having frequencies from 2700 Hz to 5400 Hz.

The energy \( E \) is calculated for each window. The energy \( E \) is defined by

\[
E = \frac{1}{N} \sum_{k=1}^{N} s^2(k)
\]

where \( E \) is the energy in the window, \( s(k) \) is the \( k \)-th sampled signal in that window, and \( N \) is the numbers of samples in the window. The current window is designed to overlap 50% with the previous window. The energy envelope of a typical passenger car is shown in Fig. 3. In Fig. 3(a), the original signals sampled with 44.1 kHz are shown. The filtered signals are shown in Fig. 3(c). In Fig. 3(b), the energy calculated in each window is shown. The average energy \( \text{av}E \) of the energy \( E \) in the current window and five previous windows is shown in Fig. 3(d). The average energy \( \text{av}E \) is used in the detection task, while the energy \( E \) in each window is used mostly in the classification task. Besides the energy \( E \) information in the whole band of frequencies, the upper band of frequencies, 4500 to 5400 Hz, is selected. Fifth-order high-pass filter filters the filtered signals used in the detection. The average energies \( \text{av}E \) of this band of frequencies are also calculated and used in the feature extraction process. Features will be extracted from both frequency bands, e.g., from 2700 to 5400 Hz and from 4500 to 5400 Hz.

![Passenger Car Information]

2.2 Vehicle Detection Algorithm

The detection algorithm was designed to detect not only one vehicle but also to initialise a feature extraction process for vehicle classification. Although different
vehicle classes may have different sizes, the detection algorithm has to overcome this problem and reduce an under count and over count rate. The under-count means that the algorithm does not detect the presence of a vehicle. It happens often to a small vehicle such as a passenger car. The over-count means that the algorithm can detect one vehicle as two vehicles. It happens often to a large vehicle such as a tractor-trailer truck.

The over count affects not only on the vehicle count information but also the classification performance. In the over count situation two different sets of features are extracted for one vehicle. These two sets of features may not be the same, as one set of features and the classifier may not classify them correctly. In the under count situation the vehicle is definitely not classified. In this study, the detection algorithm was designed to operate under normal highway traffic operations. The detection and feature extraction process is shown in Fig. 4.

There are three tests in the designed detection algorithm. All tests are designed to overcome under and over count problems. For the first test three requirements should be met. First, the average acoustic energy \( \text{avE} \) must be above a pre-set threshold. This pre-set threshold is set such that most of average energy \( \text{avE} \) of small vehicles in adjacent lane is below this threshold. Second, the average acoustic energy \( \text{avE} \) must be rising. The final requirement is that the operation mode must be in reset mode. The reset mode is the mode that the sensor is ready to detect a new vehicle.

After the first test is passed, the feature extraction process begins. The initial starting extraction point is determined by using the average energy \( \text{avE} \) information. The initial starting point should be at the point the average energy \( \text{avE} \) is across the threshold. All necessary information used in feature extraction such as the energy \( \text{avE}_2 \) in the upper band must be initially gathered at this point. Then, the process gathers information when at least 0.1 seconds after the initial starting point have passed and when at least one of following conditions is met:

- the first condition is that the average energy \( \text{avE} \) is below 125% of the previously pre-set threshold
- the second condition is that the average energy \( \text{avE} \) is decreasing
- the third condition is that the current window is more than 0.3 seconds away from the found peak

These conditions are set to make sure that the peak is found and there are few numbers of windows from the starting point. The indication of the energy movement indicating the energy rising or decreasing is calculated by the following equation:

\[
\text{Index} = \frac{1}{N} \sum_{n=0}^{N} \text{avE}(k - N + n) - \text{avE}(k - N + n - 1)
\]  

where \( \text{avE} \) represents the average energy, \( k \) the current window and \( N \) the number of window used, \( (N = 8 \) in this case).

Once the first peak is found, the location of the peak, the peak value and the number windows above the pre-set threshold from the initial point are used in the second test. If all following conditions are met, then the second test is passed. If any of them is not met, then the second test is failed and the operation is placed in the reset mode again.

The first condition is that the location of the found peak is not close to the centroid of energies of the last detected vehicle. The number of windows used in this case is calculated by how far the current peak from the centroid of energies of the last detected vehicle. This threshold value might come after the previous vehicle is classified. If the previously classified vehicle is big and very loud, then this threshold should be big. On the other hand, if the previously classified vehicle is small, then this threshold value should be small. This approach is to avoid counting one big truck as two small vehicles or two small vehicles as one big truck. The second condition is that the peak value is above 125% of the previous pre-set threshold. The final condition is that the number of windows above the pre-set threshold from the starting point is greater than \( N \) \( (N = 8 \) in this case).

After the second test is passed, the approximation for the number of windows following the peak is determined by the fuzzy logic systems. The detected vehicle is classified into 3 broad classes, small, medium and large depending on information of how far is the peak from the initial starting point and the peak value. A fuzzy logic system is used as the number of windows approximator. The linguistic variables are the number of windows from the initial starting point to the peak and the peak value. This fuzzy logic system uses the singleton-fuzzifier, product inference, and, the proposed
in this paper, balance of area defuzzifier (BOA). Once the number of windows following the peak is determined, a new peak and its location are determined again from the initial starting point to the current point. If the new and old peaks are the same, then go to the next process, centroid location of energy calculation. If the new peak is different from the old one, then, the number of windows after the peak (Nwd-After) is approximated again by using the same fuzzy logic method. After this process is done, the centroid location of energy is calculated by the following equation:

\[
L = \frac{\sum_{k=1}^{N} kE(k)}{\sum_{k=1}^{N} E(k)}
\]

where \(L\) represents the approximated location of the centroid of energy from starting point to the current position, \(N\) the number of windows from the starting point to the current one and \(E(k)\) the energy of the \(k\)th window. This location serves as the focal point of the detected vehicle’s energies.

In the third test, the centroid location is used. If this location is far from the centroid of energies of the last detected vehicle by more than 112.5% of the number of windows after the peak (Nwd-After) of the last detected vehicle, then the third test is passed. If this condition is not met, then the third test is failed. If the test is failed, then, the operation mode is in the reset mode and the detected signals are determined as parts of the last detected vehicle. If the third test is passed, then the feature extraction process begins. First, the starting and ending points are determined. The starting point is set at the point that is before the centroid point with the fixed number of windows. In this case, there are 36 windows before this centroid point for every type of vehicle. The ending point is the point that is far from the centroid point by the same number of windows as the number of windows after peak (Nwd-After). In this way, a different ending point, depending on its peak value and how far the peak is from the starting point. Feature extraction will be presented in the next section. Once the features are extracted, then the vehicle can be classified. There are numerous features that can be extracted. At this time, only features involving the energy envelope in time domain have been considered and using experts’ knowledge, 30 features have been extracted.

### 3. Classifier Design

Existing methods are examined and used in comparison to new proposed approach. There are two major paradigms studied: multilayer perceptrons (MLP) and adaptive fuzzy logic systems (AFLS).

#### 3.1 Multilayer Perceptrons

The multilayer perceptron (MLP) is the most widely used paradigm among neural networks in various applications [6]. Even if it is far from the real biological neural system, it is a powerful signal-processing algorithm for solving many practical problems. It is known to be a universal approximator in which it can approximate any function to any degree of accuracy given enough nodes in hidden layers. Although MLP networks can be trained by several methods, in this study the traditional technique of gradient-descent method is adopted.

#### 3.2 Adaptive Fuzzy Logic Systems

An adaptive fuzzy logic system (AFLS) is a fuzzy logic system having adaptive rules. Its structure is the same as a normal FLS but its rules are derived and extracted from given training data. In other words, its parameters can be trained like a neural network approach, but with its structure in a fuzzy logic system structure [7]. Since we have general ideas about the structure and effect of each rule, it is straightforward to effectively initialise each rule. This is a tremendous advantage of AFLS over its neural network counterpart. The AFLS is one type of FLS with a singleton fuzzifier and centre average defuzzifier. The centroid defuzzifier cannot be used because of its computation expense and that it prohibits using the error backpropagation-training algorithm. The proposed AFLS consists of two defuzzification approaches, centre average (CA) and a new defuzzification approach, balance of area (BOA). Each approach is derived, compared and tested in the classification problem.

##### 3.2.1 MIMO AFLS with CA Defuzzification

This AFLS has the same approach as the system presented by Wang [8]. The only difference in the structure is that the proposed AFLS is multi-input and multi-output (MIMO). The MIMO AFLS has the following rule form:

\[
\text{IF } x_1 \text{ is } c_1, x_2 \text{ is } c_2, ..., x_n \text{ is } c_n \text{ THEN } y_1 \text{ is } o_1, y_2 \text{ is } o_2, ..., \text{ and } y_p \text{ is } o_p.
\]

The proposed MIMO-AFLS has a singleton fuzzifier, product-inference engine and CA defuzzifier. It has a feedforward structure with an extra “fuzzy basis” layer. The fuzzy basis layer consists of fuzzy basis nodes for each rule. A fuzzy basis node has the following form:

\[
^m \mu (\bar{x}) = \frac{1}{\sum_{l=1}^{L} \mu^m_l (\bar{x})}
\]

where \(^m \mu (\bar{x})\) is a fuzzy basis node for rule \(m\) and...
The combination of the weighted outputs of the fuzzy basis, \( m \) respectively, of the membership function where \( \text{id} \) is in the following form:

\[
m(x) = \prod_{i=1}^{n} \tilde{I}_{F_i}^{m}(x_i)
\]

where \( \tilde{I}_{F_i}^{m}(x_i) \) is a membership value of the \( i^{th} \) input of rule \( m \). For this specific problem, a Gaussian shape has been used as a membership function of each input of each rule, then, \( \tilde{I}_{F_i}^{m}(x_i) \) will be in the following form:

\[
\tilde{I}_{F_i}^{m}(x_i) = \exp\left[ -\frac{(x_i - c_i^{m})^2}{2(b_i^{m})^2} \right]
\]

where \( c_i^{m} \) and \( b_i^{m} \) are the centre and spread parameters, respectively, of the membership function \( i^{th} \) input of the \( m^{th} \) rule. Thus, the output of AFLS is a linear combination of the weighted outputs of the fuzzy basis, is in the following form:

\[
O_p(\bar{x}) = \sum_{m=1}^{M} y_{pm} \tilde{I}_m(\bar{x})
\]

where \( y_{pm} \) is interpreted as the centre of the membership function of the \( p^{th} \) output of the \( m^{th} \) rule. In this form, the AFLS is very similar to radial basis function networks (RBFN).

### 3.2.2 MIMO AFLS with BOA Defuzzification

The most popular defuzzification method is the centroid calculation that returns the centroid of the area formed by the consequent membership function, the membership value of its rules and the max-min or max-product inference. In the case of a discrete universe, the centroid calculation yields

\[
y = \frac{\sum_{q=1}^{Q} \tilde{I}_m(y_q) y_q}{\sum_{q=1}^{Q} \tilde{I}_m(y_q)}
\]

where \( Q \) is a number of quantisation levels of the output. The higher \( Q \) is, the finer \( y \) will be. The computation will increase as \( Q \) increases as a trade off, thus this method is very computationally expensive, so that it is not appropriate to use this method in AFLS. In other words, it will be very computationally expensive to make this method adaptive. Since this method provides good performance, its main characteristics, centre of gravity and use of the shape of membership function, will be preserved in the design of a new defuzzification approach. The overall output of the system may be the result of fuzzy union or the addition of rule outputs as in Kosko’s method. The following AFLS will use Kosko’s method with product inference [9]. In general form, the calculation of the output, \( y \), will be

\[
y_p = \frac{\sum_{m=1}^{M} \tilde{I}_m L_p^m y_p^m}{\sum_{m=1}^{M} \tilde{I}_m L_p^m}
\]

where \( y_p \) : the \( p^{th} \) output of the network \( m \) : the membership value of the \( m^{th} \) rule. \( L_p^m \) : the spread parameter of the membership function in the consequent part of the \( p^{th} \) output of the \( m^{th} \) rule. \( y_p^m \) : the centre of the membership function in the consequent part of the \( p^{th} \) output of the \( m^{th} \) rule. The centre average defuzzifier was modified such that the spread parameter was included into the equation by using common sense. The AFLS with this proposed new defuzzification method is illustrated in Fig. 5.

![AFLS with BOA defuzzification method](image)

**Fig 5: AFLS with BOA defuzzification method**

Training Procedure for MIMO-AFLS with BOA Defuzzifier is similar with the one with CA defuzzification and neural networks. By using an error-backpropagation technique, all parameters have been updated during the training phase of the network. The initialisation methods in the CA defuzzification can also be used in this approach. Since the desired output in the classification problem is primarily a binary output representing each class, therefore, the initial spread parameter, \( L_p^m \), can be set to 0.75. This method can be interpreted that we have initially equal confidence in each rule. This spread parameter will be adjusted during training.

### 4. Classification of Vehicle Data

Training and testing data used in this research are collected from the roadside sensor station and processed using the described detection and feature extraction. The whole data consists of five classes. Class 1 includes passenger cars. Class 2 includes pickups, vans and mini-vans. Class 3 includes all single unit trucks having two
axles, six tires. They are small trucks, small trucks with flatbeds, small trucks with flatbeds including loads, trash trucks, heavy-duty trucks, trucks with box, etc. Class 4 includes all single unit trucks having three axles, ten tires. They are heavy-duty trucks, trash trucks, cement trucks, trucks with box, etc. The last class, class 5, includes all five-axle single trailer trucks that include tractor-trailer trucks, gas trucks, flatbeds, and flatbeds with load. Class 1 has the most dominant number in the whole data and class 4, the fewest. Because of the weather during the data-collecting period, there are few numbers in class 4. These data were collected in various conditions, from cold to warm weather, from cloudy to clear sky and from early morning to late afternoon. The road surfaces are from dry to light wet surface. Data during snow and raining are excluded. All data are collected under normal highway traffic operations. Vehicle acoustic data were collected, processed, and extracted from raw acoustic data as explained in the previous sections. There are 30 extracted features for each vehicle. There are data representing 2440 passenger cars, 1007 pickups or vans, 587 two-axle six-tire trucks, 309 three-axle single unit trucks, and 963 five-axle single trailer trucks in this data set. In a four-class study, classes 1 and 2 are combined as a small vehicle class with no change in the remaining classes. In a two-class study, class 1 and 2 are combined as a small vehicle class and classes 3 to 5 are combined as a large vehicle class. Training data is randomly picked, 75% of each class from the entire data, and the remaining data is used as the testing data.

4.1 Five-Class Problem
The five-class problem is the toughest problem in this study. Each class consists of a variety of vehicles. For example, the two-axle six-tire class includes all small trucks having two axles and six tires. This class overlaps with other classes, e.g., pickups, heavy-duty trucks. The classifier learned well in certain classes, and it would definitely classify well in those classes. The numbers of MLP hidden nodes are determined by an increment method procedure. The network starts with a small number of hidden nodes then it is incremented until the outputs of the network saturated. The final number of hidden nodes is determined by the smallest network giving relatively the same results as a bigger one. With the AFLS the numbers of rules are determined by a similar increment method procedure. The initial parameters of both AFLSs are initialised by the training data picked randomly. The input feature vector is used as the initial centres of the antecedent part. The centres of the consequent part are initialised by the corresponding desired output. In classification problems the desired outputs are normally in binary form. The Gaussian shape function is used as a membership function in both AFLS.

<table>
<thead>
<tr>
<th>% Class</th>
<th>Cl. 1</th>
<th>Cl. 2</th>
<th>Cl. 3</th>
<th>Cl. 4</th>
<th>Cl. 5</th>
<th>Total correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>74.75</td>
<td>67.46</td>
<td>72.11</td>
<td>81.82</td>
<td>90.46</td>
<td>76.34</td>
</tr>
<tr>
<td>CA</td>
<td>80.00</td>
<td>61.90</td>
<td>72.11</td>
<td>70.13</td>
<td>91.70</td>
<td>77.24</td>
</tr>
<tr>
<td>BOA</td>
<td>83.11</td>
<td>59.13</td>
<td>75.51</td>
<td>85.71</td>
<td>93.78</td>
<td>79.80</td>
</tr>
</tbody>
</table>

Table 1

The results show that the AFLS-BOA is more consistent and have better performance than the MLP and the AFLS-CA , as shown in Table 1.

4.2 Four-Class Problem
In a four-class problem, Classes 1 and 2 are combined as one small vehicle class. The remaining classes are the same as in previous problem. The number of rules is determined with the same procedures as in the previous problem. The number of rules is 14. In this problem the numbers of data are 3447, 587, 309, and 963 for Class #1, #2, #3, #4, respectively. Class 1 “dominates” the other classes. Again, the AFLS-BOA has better results than the other classifiers as illustrated in table 2.

<table>
<thead>
<tr>
<th>% Class</th>
<th>Cl. 1</th>
<th>Cl. 2</th>
<th>Cl. 3</th>
<th>Cl. 4</th>
<th>Total correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>93.62</td>
<td>73.47</td>
<td>87.01</td>
<td>88.38</td>
<td>90.05</td>
</tr>
<tr>
<td>CA</td>
<td>93.85</td>
<td>81.63</td>
<td>74.03</td>
<td>91.70</td>
<td>90.96</td>
</tr>
<tr>
<td>BOA</td>
<td>95.59</td>
<td>79.59</td>
<td>79.22</td>
<td>92.12</td>
<td>92.24</td>
</tr>
</tbody>
</table>

Table 2

4.3 Small versus Large Vehicle Classification
In this problem vehicles in Classes 1 and 2 are combined as a small vehicle class and Classes 3, 4, and 5 are combined as a large vehicle class. There will be 3447 in one class and 1859 in the other class. Again the AFLS-BOA has better classification performances than the AFLS-CA and MLP. Table 3 shows the related results.

<table>
<thead>
<tr>
<th>% Class</th>
<th>Cl. 1</th>
<th>Cl. 2</th>
<th>Total correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>97.56</td>
<td>95.70</td>
<td>96.91</td>
</tr>
<tr>
<td>CA</td>
<td>97.45</td>
<td>97.20</td>
<td>97.36</td>
</tr>
<tr>
<td>BOA</td>
<td>97.80</td>
<td>97.20</td>
<td>97.59</td>
</tr>
</tbody>
</table>

Table 3

5. Conclusions
This paper involves classification system design for vehicle acoustic signal classification. A detection algorithm was developed to detect a presence of a passing vehicle. This detection algorithm initialised and ended the feature extraction process. All features were time domain features extracted directly from acoustic energy information. In classifier design MLPs and AFLSs were used because of their trainability and
generalisation. MIMO-AFLSs were developed in contrast to the normally decomposed MISO-AFLSs for each output. There are two MIMO-AFLSs designed in this paper. The first system consists of singleton fuzzifier, fuzzy rule base, fuzzy inference engine, and centre average defuzzifier. The second system consists of the same components except a different defuzzifier, balance of area defuzzifier. The BOA uses the shape information of fuzzy membership functions in the consequence part of the IF-THEN rules to obtain the result. Its output is close to the centroid of area defuzzification (COA) while requiring much less computation.

The advantage of using an AFLS over a MLP is that its parameters can be initialised more effectively than MLPs. With good initialisation the AFLS trains much faster than a corresponding MLP.

The AFLS-BOA is a very effective system; its results are better than the standard AFLS-CA based classifier. It demonstrated possibility of an acoustic sensor system with 97.59% correct classification rate between small and large vehicles on 1327 vehicles, 92.24% correct classification rate in four-class problem on 1327 vehicles, and 79.80% correct classification rate in five-class problem on 1327 vehicles.

References: