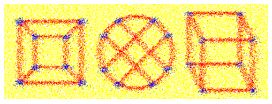
Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 11, no. 1, pp. 277–307 (2007)

# Specific Selection of FFT Amplitudes from Audio Sports and News Broadcasting for Classification Purposes

*Marios Poulos* *George Bokos*
*Nikolaos Kanellopoulos* *Sozon Papavlasopoulos*

Department of Archives and Library Sciences
Ionian University
http://www.ionio.gr/
mpoulos@ionio.gr gbokos@ionio.gr kane@ionio.gr sozon@ionio.gr

*Markos Avlonitis*

Department of Informatics
Ionian University
http://www.ionio.gr/
avlon@ionio.gr

**Abstract**

In this paper we investigate the problem of classification between sports and news broadcasting. We detect and classify files that consist of speech and music or background noise (news broadcasting), and speech and a noisy background (sports broadcasting). More specifically, this study investigates feature extraction and training and classification procedures. We compare the Average Magnitude Difference Function (AMDF) method, which we consider more robust to background noise, with a novel proposed method. This method uses several spectral audio features which may be considered as specific semantic information. We base the extraction of these features on the theory of computational geometry using an Onion Algorithm (OA). We tested the classification procedure as well as the learning ability of the two methods using a Learning Vector Quantizer One (LVQ1) neural network. The results of the experiment showed that the OA method has a faster learning procedure, which we characterise as an accurate feature extraction method for several audio cases.



| Article Type | Communicated by | Submitted | Revised |
|---|---|---|---|
| Regular paper | G. Liotta | December 2006 | May 2007 |

# 1 Introduction

## 1.1 Specific Objectives

This study investigates the problem of classifying two different files with highly similar audio overlapping regions, these being a sports broadcast and a news broadcast. The classification problem focuses on the development of a number of features extracted in order to bring out the differences of these two examples, and simultaneously to downgrade the similarity of the audio features. We introduced a new method to do this which is based on an onion algorithm (OA) of computational geometry; this reduces the number of fast Fourier transform (FFT) amplitudes of an audio signal, holding the smallest layer, which, according to latest studies [5, 24, 23, 22, 29, 27, 26, 25], encloses a dominant part of the semantic information of the signal. Thus, the objective of this study is to verify the above claim with a well-conducted experiment corroborating this technique. To implement this experiment we selected the best feature extraction, which is used for the same classification purposes [35] as the Average Magnitude Difference Function (AMDF) method, and we compared this with the proposed algorithm using as an unbiased criterion the well-fitted artificial Learning Vector Quantize (LVQ) neural network.

## 1.2 General Background

Video is a rich source of information, with visual, audio, and textual content. Many applications, such as information indexing and retrieval in multimedia databases, video editing, and so forth, require video scene analysis and classification. Research in this area in the past several years has focused on the use of speech and image information [34, 16, 17, 33, 32]. A large number of useful features, based on video and audio, have been proposed for video classification. Specifically, the foundation of any type of audio analysis algorithm is the extraction of numerical feature vectors that characterise the audio content. Until now, feature extraction has been based on a variety of feature sets. These are Time Domain features, such as ZeroCrossings, Root-Mean-Squared Energy (RMS) and Ramp Time, the Spectral Domain features Centroid, Rolloff, and Flux, Mel-Frequency Cepstral Coefficients (MFCC), and linear Predictive Coefficients (LPC). More details about the definitions of these features can be found in Hauptmann and Witbrock (1997) and Tzanetakis and Cook (2002) [8, 36]. The latest studies performed the pitch calculation using the AMDF method [10], which proved to be more robust to background noise and music in comparison with the above methods. In addition, according to Tzanetakis and Chen's (2004) recent findings [35], the MFCC and LPC features did not perform as well as the pitch calculation using the AMDF method, probably because these features are designed for speech modelling and recognition and don't work as well for modelling more general audio textures [35]. The Problems: In

this study we investigated the classification problem of two multimedia types of audio broadcasting programs. These were recordings of football and basketball matches and an audio news broadcasting file. The proposed algorithm attempts to improve the feature extraction technique in a novel way in order to eliminate the classification problems that are present in the literature. To implement this, we studied the problems as sourced from the calculation of the coefficients, which we extracted using the AMDF method, the most robust of the feature extraction techniques. In the literature these problems are the overlapping of similar backgrounds sounds, the criterion of the audio signal segmentation, and the criterion of the estimation of the interval frame which is needed for pitch calculation. More details of these problems are presented below.

## 1.3 The Problems

In this study we investigated the classification problem of two multimedia types of audio broadcasting programs. These were recordings of football and basketball matches and an audio news broadcasting file. The proposed algorithm attempts to improve the feature extraction technique in a novel way in order to eliminate the classification problems that are present in the literature. To implement this, we studied the problems as sourced from the calculation of the coefficients, which we extracted using the AMDF method, the most robust of the feature extraction techniques. In the literature these problems are the overlapping of similar backgrounds sounds, the criterion of the audio signal segmentation, and the criterion of the estimation of the interval frame which is needed for pitch calculation. More details of these problems are presented below.

## 1.4 The Problem of Overlapping Background Sounds

Sound recordists know that the audio in a sport-broadcasting video is different from that in a news report. However, the main problem becomes focused when the two categories (sport audio and news audio) overlap heavily in the same region, such as when the background sounds are similar. This can be due to incorrect or noisy training labels. These problems of the inconsistency and incompleteness of human annotations are so prevalent that any video classification systems must cope with them [35]. It will be difficult for discriminative classifiers to make these distinctions, because there is no clear, decisive boundary separating the two sets of data. A solution to the above problems is selecting a suitable length for each shot, in particular by obtaining the decision for the whole shot by the majority of classified windows within it and using the percentage of this majority of windows as a confidence measure for classification. This approach has the advantage of dealing elegantly with the problem of shots that contain two different audio textures [6], which, although uncommon, occurs sometimes in the data. On the other hand, this solution increases the complexity dramatically, which is quite ineffective for Moving Picture Experts Group Seven (MPEG7) [1].

## 1.5 The Problem of Audio Time Segmentation

The foundation of any type of audio analysis algorithm which is based on the extraction of numerical features needs the determination of the audio time segmentation. However, all the techniques (including the AMDF) of audio features extraction based on the variable duration shot [35, 16, 3] usually range between one and six seconds. The determination of this time segmentation and its particular sub-segmentation are highly significant, as they take place using a specific overlapping window called an interval frame.

## 1.6 The Estimation of the Interval Frame

The other problem, which derives from the previous procedure, is the determination of the interval frame. This problem depends on the voice's features . For example, a speaking voice requires a variable interval frame ranging between 2.3 ms and 15.9 ms. However, in the non-speech intervals within speech, or breathing pauses, we need a variable segmentation ranging between 100-300 ms [20, 31]. In our example, this problem is serious for the existing methods, as the dominant feature of the classification is most often the background activity, which is a non-speech signal. Thus, we selected an interval time for the AMDF coefficients greater than 50 ms in order to include all the examples in our experiment (see section 3.1).

## 1.7 The Proposed Method

The primary idea is to extract an audio feature that attempts to avoid noise effects by not using the vulnerable parts of speech spectra and without losing important discriminative information. This approach differs from noise removal methods, such as the AMDF, because it does not require an estimate of the noise and does not assume a stationary or slowly changing noise. The solution to this problem is the reduction of the spectral resolution of the original FFT amplitudes using the OA, according to our latest studies [26, 25]. The basis of the method depicts the centre of the multi-layers of a set of arithmetic points on the Cartesian plane that represent the values of the application used. For example, in the case of fingerprint verification [26], these values represent the values of the FFT amplitudes, which are produced by the pixels' values. In our example, these values represent the FFT amplitudes of the original audio signal. Similarly, this algorithm can be applied in a text categorization procedure [25]. In this technique, however, we replaced the FFT method with a numerical conversion of the text characters, thus testing the proposed method for the first time in the audio signals area in order to ascertain its ability to classify the two different audio categories. Finally, the present work focuses in principle on sports broadcasting as opposed to audio news broadcasts and aims to establish a one-to-one correspondence between the specific information and certain appropriate features of each audio signal category. A neural network classifier, Learning Vector Quantizer (LVQ), is employed to classify unknown

features of each example from the AMDF method in comparison with the OA method in order to show the OA method's classification superiority over the classic AMDF-one Neural network-based classification, which has received considerable attention recently in a wide variety of research fields and experimental setups. The specific type of neural network employed here, namely the LVQ, offers the advantage of classifying input vectors of high dimensionality, which is desirable for our tests. A more detailed description of its architecture and operation is provided in Section 2.3. Spectral values obtained from both methods (OA and AMDF) are used as features to form the input vectors. Furthermore, we test the validation of the feature vector of each method in the training procedure by investigating the training error of convergence of each. This approach is called the hold out method [30].

# 2 Method

## 2.1 Overview

The present study is divided into a feature extraction stage and a training and classification stage. In the feature extraction stage, the OA method is based on a novel statistical estimation in which the smallest layer of an onion convex polygon encloses the geometric median value of a feature vector [10]. Furthermore, this statistical approximation has been verified empirically in several pattern recognition problems [24, 23, 22, 29, 27, 26, 25]. In our example, the feature vector is composed by the FFT amplitudes of a particular shot of audio file of either sport or news video. Specifically, we will use the Matlab function *fft*$(x)$ to do a Fourier Analysis of the data. This is the discrete Fourier transform (DFT) of vector $x$, computed with an FFT algorithm. If $X$ is a matrix, *fft*$(x)$ is the FFT of each column of the matrix. In Matlab, all variables are matrices; vectors are simply row or column matrices. The *fft* function employs a radix-2 Fourier transform if the length of the sequence is a power of two, and a slower mixed-radix algorithm if it is not. The function implements the transformation given by the following equation (1):

$$X(k+1) = \sum_{n=0}^{N-1} x(n+1) e^{-ik\frac{2\pi n}{N}} \tag{1}$$

where $N = length(x)$. Note that the series is written in an unorthodox way, running over $n+1$ and $k+1$ instead of the usual $n$ and $k$, because Matlab vectors run from 1 to $N$ instead of from 0 to $N-1$.

In our example, in the Cartesian plane (see Figure 1) the absolute values of $X$ are in the $y$ axis and in the $x$ axis are the order of each element of matrix $X$. Thus, we constructed a vector matrix $\mathbf{S}$ of size $(N \times 1)$

$$\mathbf{S} = \begin{bmatrix} |X_1| \\ . \\ . \\ . \\ |X_N| \end{bmatrix}$$

The OA method is described as follows:

1. We put the elements of the **S** vector in the Cartesian plane according to $f(N, |X_N|)$ function. For example see Figure 2.

2. We determine the finite set of points $\mathbf{S} = \mathbf{S}_0$. Let $\mathbf{S}_1$ be the set $\mathbf{S}_0 \backslash \partial H(\mathbf{S}_0)$ : **S** minus all the points on the boundary of the hull of **S**. (see figure 3)

3. The process continues until reaching a set with three points or less. Similarly, define $\mathbf{S}_{i+1} = \mathbf{S}_i \backslash \partial H(\mathbf{S}_i)$ . The hulls $H_i = \partial H(\mathbf{S}_i)$ are called the layers of the set and the process of peeling away the layers is called onion peeling [19, 7]. (see figure 4)

This position may be determined by using a combination of computational geometry algorithms, which is known as Onion Peeling Algorithms [4], with overall complexity $O(d \cdot n \log n)$ times, where $d$ is the depth of the smallest convex layer and $n$ is the number of characters in the numerical representation (in accordance with section 2.1).

Thus, the smallest convex layer $\mathbf{S}_i$ of the original set **S** of vector carries specific information. In particular, vector $\mathbf{S}_i$ may be characterized as a common

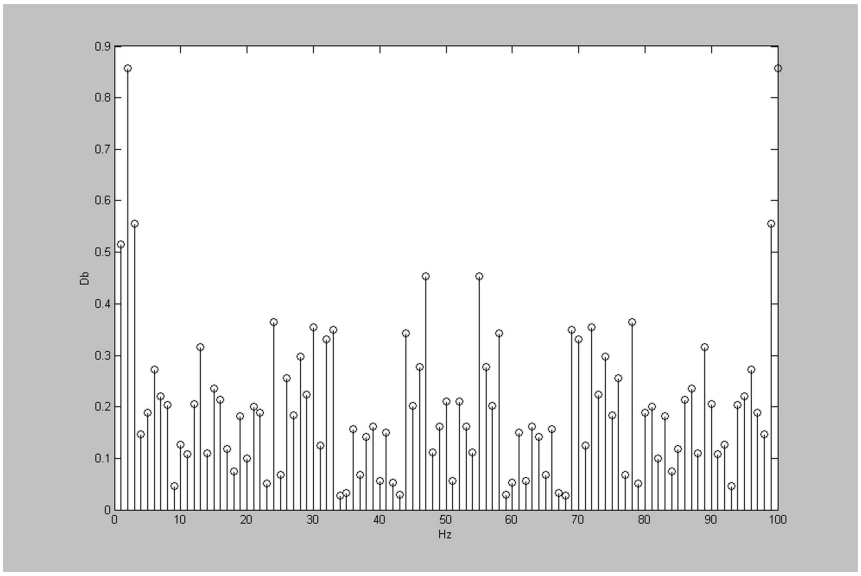


Figure 1: The placement of vector **S** in Cartesian plane.

geometrical area of all the elements of vector **S**. In our example, this consideration is valuable because this subset may be characterized as representing the significant semantics of the selected audio signal (see figure 5). The decision

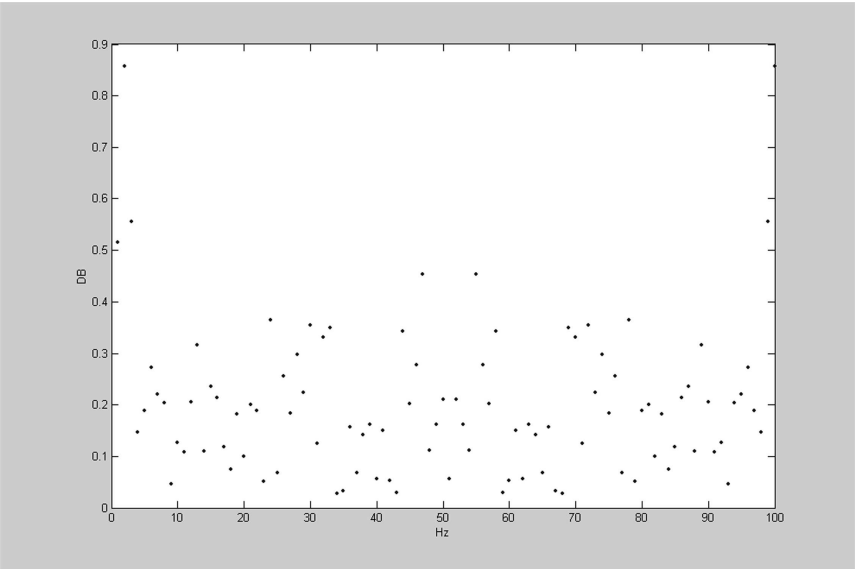


Figure 2: The placement of coordinates $f(N, |X_N|)$.

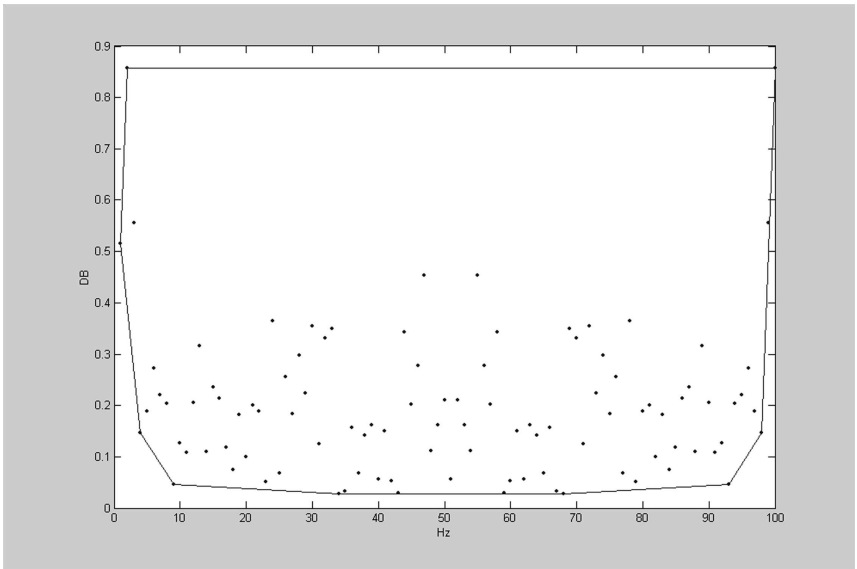


Figure 3: The external hull $\mathbf{S}_0$

regarding this selection is explained in the experimental section.

We may consider the smallest convex layer to comprise a significant geometrical region of frequency enclosing the median frequencies of the original

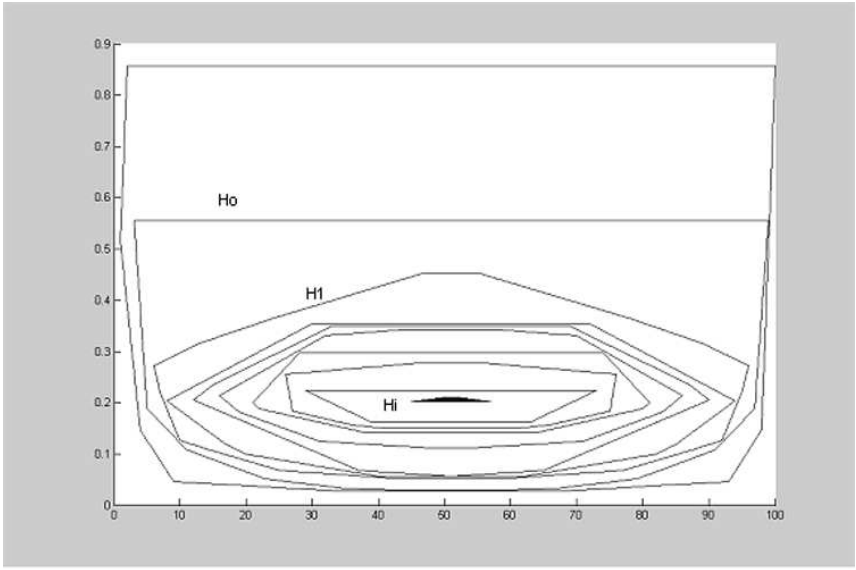


Figure 4: The iterative procedure of convex hulls

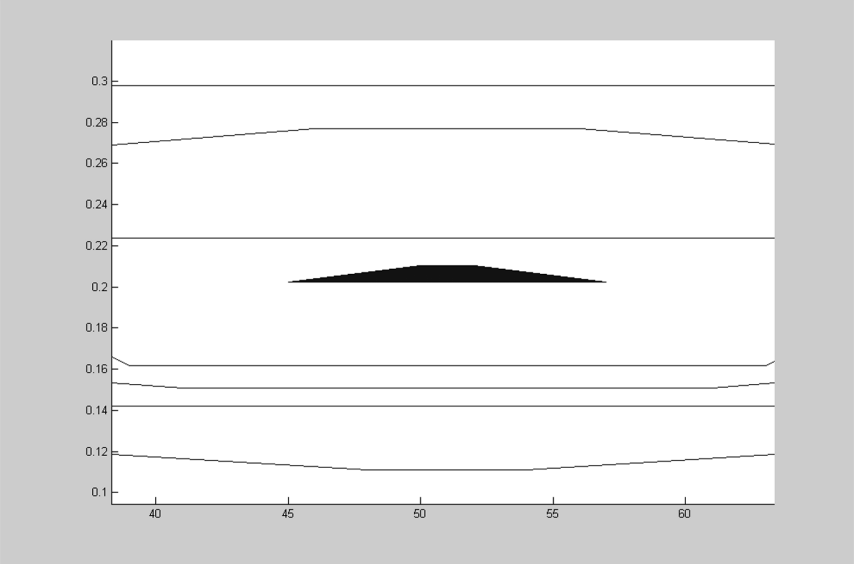


Figure 5: The isolation of the smallest convex polygon

audio shot file. The advantage of this method over the AMDF method is that the problems of length selection (shot and pitch period) are eliminated, as this method avoids the problem of specific frequency segmentation and creates coefficients from the original FFT amplitudes. These, in turn, have the additional advantage of being appropriate for application to an inverse Fourier transform in order to a particular signal from the original audio file to be retrieved. The main advantage of the proposed algorithm is that it is possible to be used in a real time scenario. For the justification of this claim we constructed a scenario, which is presented in Section 6. Furthermore, we processed the same shots of audio sport or news files by the AMDF method. These extracted features coefficients consist of the AMDF features vector, which is to be used in the next stage. The training and classification procedure was determined taking into account the following criteria:

1. The selection of the appropriate neural network, which is best fitted for the classification procedure of the above examples (OA, AMDF). The justification of this selection is presented in section 2.3.

2. The selection of the optimum size of the feature vector for the well-functionality which is used in the selected neural network. More details are presented in sections 2.2.2 and 3.2.

3. The selection of the suitable number of feature vectors needed for training and testing procedures for the selected neural network in order to yield accurate results. The selection took place using bibliographic research. More details of this selection are presented in section 3.3

4. The determination of the training group of feature vectors per category that yielded the minimum error training convergence. More details of this are presented in section 3.3.

Thus, an equal number of feature vectors from both of the OA and AMDF methods respectively are trained using an independent LVQ1 neural network. The LVQ1 neural network is adopted according to the bibliographic research [15]. Specifically, in the comparison among the SVM , K-NN classifier and LVQ neural network showed that LVQ is more sensitive to the feature audio (speech data) data than the any other classifiers in the test. Furthermore LVQ yields satisfactory results for well discriminating features [15].

The remaining feature vectors of both examples are submitted to the testing procedure. The justification of this selection centres on the ability of this neural network to classify the above features better than other neural networks, because an LVQ1 codebook contains highly structured lattice points that effectively span the signal space [18]. Furthermore, we tested the learning ability of the two categories in a statistical learning error convergence procedure which we explain in the experimental section.

## 2.2 Feature Extraction Using the OA and AMFD Methods

In this stage we isolated the original audio of a sport video file in Mpeg-2 format using a suitable multiplexes program. Thereinafter, we segmented the audio

signal into shots and extracted the features in two feature vector categories: AMDF and OA. At this point, it must be noted that the configuration of the specific segmentation of all the audio files took place in order for the AMDF processing to yield the maximum classification results according to the reports in the literature. Thus, we used the same settings in the OA algorithm in order to carry out an honest comparison between our method and the most robust AMDF method. In this setting, we therefore determined the particular shots, which are ranged between one to six seconds, for the calculation of the AMDF coefficients. This selection was calculated taking into account two parameters. The features extraction should not exceed the number of 20 coefficients (see section 3.2), and the interval frame must be range between 50-300 ms (see introduction part). Thus, we obtained AMDF coefficients, which were extracted from $20 \cdot 50\ ms = 1000\ ms = 1\ sec$, $20 \cdot 100\ ms = 2000\ ms = 2\ sec$, $20 \cdot 150\ ms = 3000\ ms = 3\ sec$, $20 \cdot 200\ ms = 4000\ ms = 4\ sec$, $20 \cdot 250 = 5000\ ms = 5\ sec$ and $20 \cdot 300 = 6000\ ms = 6\ sec$ shot duration segments. Moreover, this segmentation came into agreement with the literature [24, 35, 17].

### 2.2.1 AMDF Feature Vector

The AMDF method is based on the following property: Suppose that a digital speech signal $x(n)$ is periodic with period $T$. Then the difference between two samples is determined as:

$$\mathit{Diff}(m) = x(n) - x(n+m)$$

Thus, the difference signal $\mathit{Diff}(m)$, is calculated by delaying the input speech various amounts, subtracting the delayed waveform from the original, and summing the magnitude of the differences between sample values, using the following equation (3):

$$AMDF(m) = \frac{1}{t} \sum_{n=0}^{t-1} |x(n) - x(n+m)| \tag{2}$$

$$0 \leq m \leq t-1$$

Where $n$ is the sequence number of the speech wave and $t$ is the sample number. For reasons of brevity the $m$ elements which are extracted according to Equation 2 are named AMDF coefficients or AMDF feature vector.

### 2.2.2 OA Feature Vector

The OA feature vector is extracted in the following steps. First, the spectral density is calculated from the original audio signal $x(n)$ ($N$ samples) via the Fourier transform as described previously in Equation 1. Next, the absolute FFT amplitudes (dBV) of values are put on the Cartesian plane and submitted to the onion peeling procedure. The idea is to use the convex hull [7] subroutine

recursively to extract the outmost convex hull ($H_1$) of the given points and to apply the same subroutine to extract the convex hull of the remaining inner points ($H_2$), and so forth. The program stops when the innermost convex hull contains no more than three points. The sequence of nested convex hulls is called the onion-peeling of a given set of points. This structure can be obtained in $O(H_1)+O(H_2)+...+O(H_r)$ times by using the convex hull subroutine, where $H_1$, $H_2$, $H_r$ denotes the elements of each convex layer of the onion peeling:

$$H_1 = (|f_{11}|, |f_{12}|, ..., |f_{1x}|), ..., H_r = (|f_{r1}|, |f_{r2}|, ..., |f_{rw}|) \quad (3)$$

and where $H_1$ is the external layer and $H_r$ is the internal or smallest layer. Thus, the

$$\left|\overrightarrow{f_m}\right|$$

values are re-arranged in a new vector $\mathbf{H}$ of dimensionality $(1 \times m)$

$$\mathbf{H} = [H_1, ..., H_r] = [|f_{11}|, |f_{12}|, ..., |f_{1x}|, ..., |f_{2u}|, ..., |f_{r1}|, |f_{r2}|, ..., |f_{rw}|] \quad (4)$$

It must be noted that each layer may be of a different size, which justifies the computation of layer size (convex polygon) being unpredictable and being implemented in a non-linear manner. Finally, the feature vector is selected from the $t$ last absolute amplitudes $t$ of vector $\mathbf{H}$ which are found in the region of the smallest layer $H_r$. The value $t$, for both cases, is determined in the experimental part (3.2).

## 2.3 The Feature of the Proposed Neural Network

In our work we selected and employed a neural network called LVQ1, which was proposed by Kohonen [9] as a supervised extension of the more general family of unsupervised classifiers named Self-Organizing Maps (SOMs). The training of LVQ1 is a two-step procedure. In the first step, initial positions of the class representatives (or codebook vectors) are determined in the $r$-dimensional space using standard clustering algorithms such as the $k$-Means clustering algorithm or the Linde-Buzo-Gray (LBG) algorithm, with a given number of classes. In the second step, class representative positions are iteratively updated to minimise the total classification error of the training set of vectors. To this end, codebook vectors are directed towards the data vectors of the same class and distanced from the data vectors of different classes. A Euclidean distance measure is used for calculating distances. More specifically, every time a member of the training set, feature vector $ti$, is incorrectly classified, the two codebook vectors involved, correct $rc(i-1)$ and incorrect $rw(i-1)$, are updated as follows:

$$rc(i) = rc(i-1) + a(i)[ti - rc(i-1)],$$

$$rw(i) = rw(i-1) - a(i)[ti - rw(i-1)].$$

The rate of the update, or learning rate, $a$, controls the speed of convergence and is a descending function of time for iteration index (i). The class-separating surfaces obtained in this way are nearly optimal in the bayesian sense. Different rules applied when moving (updating) class representatives during the training iteration produce different versions of the LVQ1 training algorithm. The version employed here, namely LVQ1, is chosen for its properties of quick convergence and robustness of the class representatives' positions over extended learning periods.

In our example, the architecture the LVQ1 network used to classify the OA or AMDF feature vectors is shown in Fig. 6. Input vectors of dimensionality $20 \times 1$ are weighted and fed to the first layer of neurons, known as the competitive layer (the selection of the dimensionality 20 is explained in the experimental section). These neurons compete for inputs in what we call a greedy way; hence the layer name. Four such neurons form the competitive layer in our example. The output of the competitive layer, which is a grouping of the inputs into subclasses, is fed to the second linear layer, which groups subclasses into target classes. The weights connecting the two layers take on binary values of zero or one, which merely indicate class membership and not actual weighting.

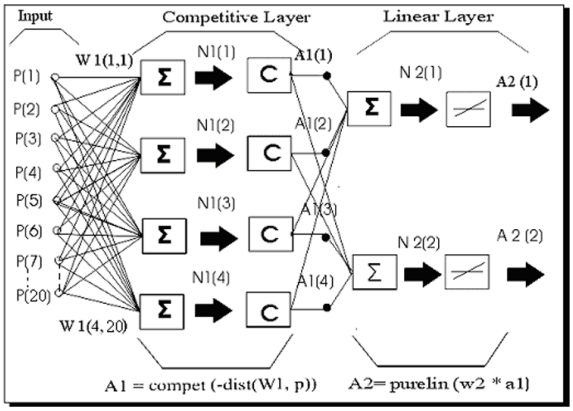


Figure 6: Architecture of the LVQ1 neural network employed for the classification for OA or AMDF input vectors of dimensionality 20.

# 3 Experimental Section

## 3.1 Experimental Data

We evaluated the proposed audio classification and segmentation algorithms by using our database, which is audio clips from TV programs (CNN, Eurosport) of news reports and football and basketball sports broadcasts. Each file contained

combinations of speech and either music or background noise (in the case of news broadcasting), and speech and noisy backgrounds (in the case of sports broadcasting). In the news reports clip, the ratio between the amount of pure speech, music, and noisy speech is about 8:1:1. In the sports clip, the ratio between the amount of pure speech and noisy speech is about 7:3. In our experiments, we set one second as a test unit, as in a similar study [17].

We obtained 70 different clips in each category (sport or news) for a total of 140 audio clips, each greater than six seconds long and sampled at 22 KHz. We used 20 for each category in training the classifier, while using the remaining 100 for testing. The 70 audio news clips were recorded from different broadcast TV programs using a monophonic audio sound configuration system. In addition to these originally selected segments we further selected segments greater than six seconds which all satisfied the aforementioned audio ratio settings. We thus created a database which contained $6 \times 70 = 420$ audio clips for each category, or a total of 840 audio clips. In the segmentation procedure we created six specific segments of one, two, three, four, five and six seconds in segmentation for each audio clip, based on bibliographic research [2, 6], in which the length of the audio clips could vary from one to six seconds. Furthermore, the data collection needed for the experimental training and testing stages was compared with a similar set on which most current research is based [16]. The proposed database of 420 audio segments proved to be a sufficient sample for our classification purposes in the statistical evaluation, which is presented in section 5.

The music content in this data set is composed mainly of environmental sound. All data are 22 kHz sample rate, mono channel and 16 bit per sample, from which we selected about 420 seconds ($1\ sec \cdot 20 + 2\ sec \cdot 20 + 3\ sec \cdot 20 + 4\ sec \cdot 20 + 5 sec \cdot 20 + 6\ sec \cdot 20 = 420\ sec$) for each category (sport or news), totalling 840 seconds, as training data, and ($1\ sec \cdot 50 + 2\ sec \cdot 50 + 3\ sec \cdot 50 + 4\ sec \cdot 50 + 5\ sec \cdot 50 + 6\ sec \cdot 50 = 1050\ sec$) for each category (sport or news), totalling 2100 seconds for testing.

## 3.2 Feature Vector Extraction

Using the 420 audio clips, the OA (figures 7, 8, and 9) and AMDF feature vectors were extracted from Equations 1, 2, and 3. Specifically, in Figures 7, 8, and 9 we can see the snapshots of the zoom of the OA analysis. In particular, in Figure 9 we can see the specific red area selection, which satisfies the criterion of 20 elements selection around the latest layer.

The value $t$ (see section 2.2.2) is determined according to bibliographic research [16, 12, 13, 11], in which the optimal dimension size depends on the experimental part in combination with the LVQ algorithms. These algorithms typically operate to preserve neighbourhoods on a network of nodes which encode the feature vector. In the scientific practice the ideal size of learning feature vector of an artificial neural network it has been determined after experimentation and concretely from the minimization of training error procedure. Thus a size of 24 elements has been showed as an optimal size [13].

Thus, we decided after experimentation that 20 elements is the optimal size

Table 1: Two examples of the calculation of the OA and AMDF feature vectors.

| *OA feature vector* | *AMDF feature vector* |
|---|---|
| 9.7306 | 0.0000 |
| 9.9551 | 0.0677 |
| 10.0127 | 0.1120 |
| 9.7790 | 0.1509 |
| 9.5094 | 0.1818 |
| 10.2665 | 0.2090 |
| 9.9895 | 0.2276 |
| 8.9618 | 0.2384 |
| 8.9007 | 0.2424 |
| 8.9036 | 0.2395 |
| 9.0546 | 0.2298 |
| 8.7992 | 0.2162 |
| 9.3747 | 0.2024 |
| 9.4351 | 0.1916 |
| 9.0578 | 0.1841 |
| 9.4087 | 0.1795 |
| 9.3630 | 0.1776 |
| 8.9171 | 0.1782 |
| 9.1252 | 0.1813 |
| 9.3524 | 0.1854 |

in our example. For better comprehension, we used an audio sports broadcasting segment of one second, which has $1 \cdot 22000 = 22000$ arithmetic samples. Using the FFT transform we submitted these values in OA processing and we received 20 central absolute values (table 1), which are contained in the region around the latest convex layer, as shown in figure 7.

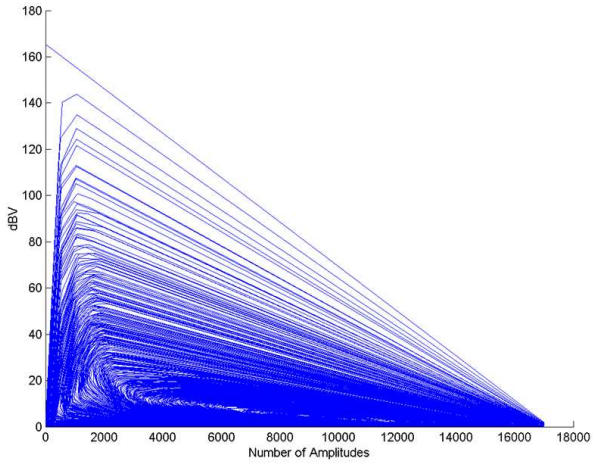


Figure 7: The visual procedure of the extraction of the Onion Algorithm.

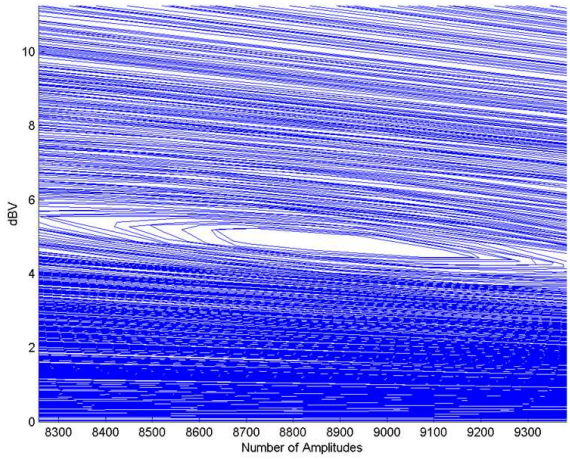


Figure 8: A visual zoom of the construction of the Onion Layer. The smallest onion layer in the centre of the figure is shown.

### 3.3 Neural Network Training and Classification Procedure

Another fundamental problem in the construction of the LVQ1 neural network is the estimation of a certain number of needed feature vectors per category in order to be able to estimate the densities accurately enough [11, 14]. These precise calculations require a large number of feature vectors, which, in practice, is not always possible. Nevertheless, pattern recognition algorithms have proven to be highly useful in this kind of small sample size problem, in which generalisation plays an important role. Much research has been done in this area [14, 21]. Considering this, we adopted 20 training feature vectors for each category (sports or news).

The next problem of the classification procedure was the selection of the 20 vectors of the original group from the original sample of feature vectors which we used for the training procedure, using the impartial hold-out method. In this method we tested each case separately (OA and AMDF methods), and all the combinations of equal numbers of vectors - in our example this number of training vectors per group was 10 (see above) - are selected as optimum selections from the group, due to yielding the minimum amount of training error quickly. In this way, we ensured that the group of each category gave the most common characteristics. In other words, we created different groups of 20 feature vectors for each category (sport and news), and repeated this procedure for all the vectors, segmented by duration (one, two, three, four, five, and six seconds). More details of this are presented in section 3.4.

Thus, we trained the two feature vectors (AMDF, OA) after experimenta-

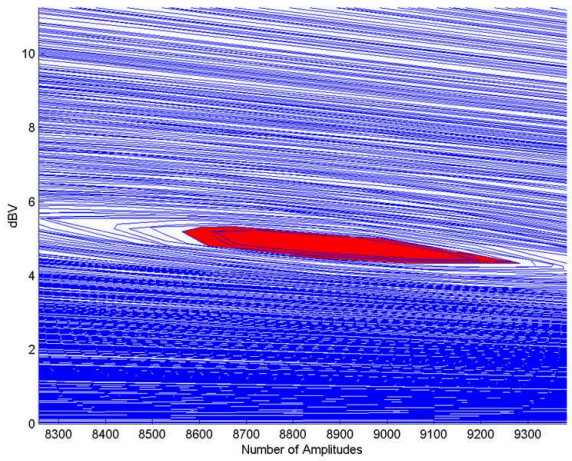


Figure 9: Feature extraction. The red area shows were the absolute FFT amplitudes were found.

tion, minimizing the data set to 40 feature vectors (20 for each category), and taking into account the convergence criterion of error epoch training (Fig.10). These feature vectors were then fed into an LVQ1 classifier [30], first for training in order to be directed towards the feature vectors of the same class and distanced from those feature vectors of a different class, and then for the actual classification of unknown input feature vectors.

### 3.4 Training the LVQ1 using AMDF and OA features

The LVQ1 neural network, which was used in the aforementioned training procedure, is described in section 2, step 2 (Feature extraction and classification), and was trained for a total of 300 cycles (epochs) with a learning rate in the order of 0.001.

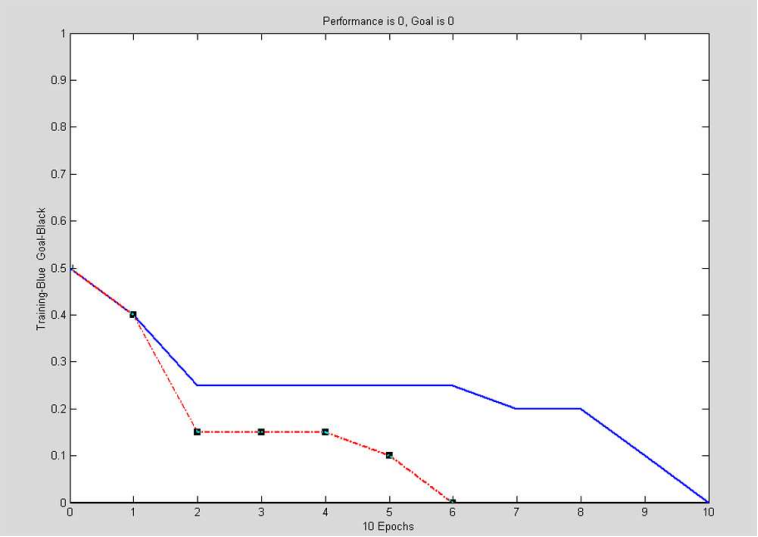


Figure 10: Error plot while training an LVQ1 network using AMDF coefficients (solid-blue line) and OA coefficients (dash dot-red line).

### 3.5 Minimum Error Training

Our goal was to find which kind of feature vector, AMDF or OA, performed better on the LVQ1 neural network. The simplest approach for the comparison of different feature vectors was to evaluate the error function, using data which was independent of that training.

We trained various feature vectors by minimizing an appropriate error function defined with respect to the LVQ1 neural network hold-out method [30]. We then compared the performance of the feature vectors by evaluating the training error function using an independent LVQ1, and selected the feature vector

having the smallest error function with respect to the LVQ1. We implemented this procedure experimentally in section 5.2.

# 4 Results

## 4.1 Classification Results

The classification results for the six tested LVQ1 neural networks are presented in tables 2, 3.

Table 2: Classification Results of LVQ1 Neural networks in time 1-3 secs.

| **Time length** | | **1 sec** | | **2sec** | | **3 sec** | |
|---|---|---|---|---|---|---|---|
| Feature Extraction Method | Classes | Sports | News | Sports | News | Sports | News |
| AMDF | Sports | **42** | **8** | **44** | **6** | **45** | **5** |
| AMDF | News | **5** | **45** | **6** | **44** | **6** | **44** |
| Sensitivity | | **0,89** | | **0.88** | | **0.88** | |
| Specificity | | **0.85** | | **0.88** | | **0.90** | |
| OA | Sports | **45** | **5** | **47** | **3** | **49** | **1** |
| OA | News | **4** | **46** | **2** | **48** | **2** | **48** |
| Sensitivity | | **0.91** | | **0.94** | | **0.96** | |
| Specificity | | **0.90** | | **0.94** | | **0.96** | |

Table 3: Classification Results of LVQ1 Neural networks in time 4-6 secs.

| **Time length** | | **4 sec** | | **5 sec** | | **6 sec** | |
|---|---|---|---|---|---|---|---|
| Feature Extraction Method | Classes | Sports | News | Sports | News | Sports | News |
| AMDF | Sports | **47** | **3** | **49** | **1** | **50** | **0** |
| AMDF | News | **4** | **46** | **2** | **48** | **1** | **49** |
| Sensitivity | | **0.94** | | **0.96** | | **0.98** | |
| Specificity | | **0.94** | | **0.98** | | **1** | |
| OA | Sports | **49** | **1** | **50** | **0** | **50** | **0** |
| OA | News | **1** | **49** | **1** | **49** | **0** | **50** |
| Sensitivity | | **0.98** | | **0.98** | | **1** | |
| Specificity | | **0.98** | | **1** | | **1** | |

As can be seen in Table 2, the AMDF method shows a weakness in correctly classifying all the cases in the first three seconds duration, while in Table 3,

in time greater than four seconds duration, the results for both methods are similar. This conclusion is reinforced by the interpretation of the extracted indices of sensitivity and specificity, which ranged between 0.85-0.90, while the OA method showed better and more successful results, specifically in that OA's indices ranged between 0.90-0.96.

According to these results, we concluded that between four and six seconds is the optimal segmentation time for both methods, but that between one and three seconds the superiority of the OA method compared to the AMDF method is obvious. More details about the data of table 4 are presented in the statistical evaluation in section 5.

### 4.2 Minimum Error Training Results

We tested the classification ability of each feature vector method in the training procedure, and investigated the error training vector set, defined as 40 feature vectors (20 for each category). Twelve (12) different LVQ1 neural networks from different vector sets (six (6) per time-duration category) were trained. The vector set was extracted from original audio files of six seconds duration. The classification results of minimum error training are presented in Table 4.

Table 4: Error training convergence in epochs while training an LVQ1 network using AMDF and OA feature vectors of 6 sec in length.

| *OA Method* | *AMDF Method* |
|---|---|
| *Number of convergence epochs* | *Number of convergence epochs* |
| 6 | 12 |
| 7 | 10 |
| 9 | 14 |
| 5 | 11 |
| 8 | 14 |
| 10 | 13 |

## 5 Statistical Evaluation

The statistical evaluation of the classification results showed that the proposed OA method is more accurate in all cases, especially in the smallest time segments (one, two, and three seconds). In all six tests (sports versus news) we considered either the true positive result or the true negative result of an input vector to be a correct classification result.

For example, as can be seen the AMDF method has a true positive recognition score for a one-second time length for the sports group of 42/50, or 84 percent, the number of true recognition cases being a=42. The true negative recognition score for the news group is 45/50 or 90 percent, the number of true

recognition cases being d=45. In the same table, for a one-second time length, it can be seen that the false positive recognition score for the sports group is 8/50 or 16 percent, the number of false recognition cases being b=8, while the false negative recognition score for the news group is 5/50 or 10 percent, the number of false recognition cases being c=5.

Consequently, we can calculate the sensitivity and specificity values of the results in table 3, which are statistical indices usually utilised in similar classification problems [28]. For example, for the AMDF method with a time length of one second, the values are calculated as follows:

$$Sensitivity = \frac{a}{a+c} = 0.89$$

$$Specificity = \frac{d}{b+d} = 0.85$$

In the same way, all the correct negative or positive classification scores are shown for both methods, along with the calculated values of sensitivity and specificity. Furthermore, the superiority of the OA method over the AMDF method is evident from the results of the minimum error training procedure.

More specifically, for all LVQ1-trained OA sets of feature vectors, the mean value of epochs of training error convergence was approximately seven, with a variance of 2.44, while in the case of AMDF this value was 12, with a variance of 2.70.

However, for further statistical processing, in order to evaluate the statistical significance of the classification scores we obtained in the experimental section, we applied the chi-square test to the results. We also compared the two feature extraction methods presented in terms of their Cramer coefficient of mean square contingency, 1. As we can see, the results are statistically significant at the a = 99.5% level of significance, and can be placed into a two-way contingency table which is structured on the basis of two criteria along its two dimensions.

Here we use 'subject belongs to class $i$' as the first criterion (vertical dimension) and 'subject is classified into class $j$' as the second criterion (horizontal dimension). An ideal classification method should produce a diagonal matrix of classification scores ('subject belongs to class $i$' and 'subject is classified into class $i$') corresponding to full dependency between the two above criteria, while practical methods would tend toward this behaviour.

Evaluation of the statistical significance of the classification results is thus transformed into a hypothesis-testing problem. The null hypothesis of the independence of the two criteria is tested against the alternative hypothesis of dependence. The test statistic used for this purpose is the $\chi^2$. Statistically significant classification results correspond to rejection of the null hypothesis at a satisfactory level of significance.

Let the contingency matrix $S$ be of dimensions $(r \times c)$, meaning $r$ rows and $c$ columns, and let the $(i,j)$-th entry of $S$, $S(i,j) = f_{ij}$ ; $i = 1, \ldots, r$; $j = 1, 2, \ldots, c$ denote observed frequency of occurrence of the event $(i,j)$ ('subject belongs to

class $i$' and 'is classified into class $j$') and $e_{ij}; i = 1, \ldots, r; j = 1, 2, \ldots, c$ denote the expected frequency of occurrence of the event $(i, j)$. Then the test statistic is given by

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(f_{ij} - e_{ij})^2}{e_{ij}} \tag{5}$$

which asymptotically follows the distribution with $(r-1)(c-1)$ degrees of freedom. When unknown, expected frequencies can be estimated from $S$ using

$$e_{ij} = \frac{R_i C_j}{N} \tag{6}$$

where $N$ is the total number of events in $S$, $R_i$ is the sum across the $i$-th row of $S$ and $C_j$ is the sum across the $j$-th column of $S$.

The degree of dependence between the two criteria can also be measured by the Cramer coefficient [37] of mean square contingency,

$$\phi_1 = \sqrt{\frac{\chi^2}{N \min(r-1, c-1)}} \tag{7}$$

Coefficient $\phi_1$ takes on values between 0 (independence) and 1 (full dependence). Two classification methods can be compared as to the statistical significance of their results in terms of their Cramer coefficient. Note that for $2 \times 2$ contingency tables, 7 becomes

$$\phi_1 = \sqrt{\frac{\chi^2}{N}} \tag{8}$$

In our example we tested the results using the above statistical criteria in six cases (one second to six seconds). In all tests, the results form $2 \times 2$ contingency tables are presented on Tables (5-10).

Table 5: Test case, subject sport versus group news classification scores based on AMDF and OA feature vectors from one-second duration.

| Classes | AMDF Method | | OA Method | |
|---|---|---|---|---|
| | Sports | News | Sports | News |
| Sports | $\frac{42}{50} = (84\%)$ | $\frac{8}{50} = (16\%)$ | $\frac{45}{50} = (90\%)$ | $\frac{5}{50} = (10\%)$ |
| | $[23, 5]$ | $[26, 5]$ | $[24, 5]$ | $[25, 5]$ |
| News | $\frac{5}{50} = (10\%)$ | $\frac{45}{50} = (90\%)$ | $\frac{4}{50} = (8\%)$ | $\frac{46}{50} = (92\%)$ |
| | $[23, 5]$ | $[26, 5]$ | $[24, 5]$ | $[25, 5]$ |

Table 6: Test case, subject sport versus group news classification scores based on AMDF and OA feature vectors from two-seconds duration.

| Classes | AMDF Method | | OA Method | |
|---|---|---|---|---|
| | Sports | News | Sports | News |
| Sports | $\frac{44}{50} = (88\%)$ | $\frac{6}{50} = (12\%)$ | $\frac{47}{50} = (94\%)$ | $\frac{3}{50} = (6\%)$ |
| | $[25, 0]$ | $[25, 0]$ | $[24, 5]$ | $[25, 5]$ |
| News | $\frac{6}{50} = (12\%)$ | $\frac{44}{50} = (88\%)$ | $\frac{2}{50} = (4\%)$ | $\frac{48}{50} = (96\%)$ |
| | $[25, 0]$ | $[25, 0]$ | $[24, 5]$ | $[25, 5]$ |

Table 7: Test case, subject sport versus group news classification scores based on AMDF and OA feature vectors from three-seconds duration.

| Classes | AMDF Method | | OA Method | |
|---|---|---|---|---|
| | Sports | News | Sports | News |
| Sports | $\frac{45}{50} = (90\%)$ | $\frac{5}{50} = (10\%)$ | $\frac{49}{50} = (98\%)$ | $\frac{1}{50} = (2\%)$ |
| | $[25, 0]$ | $[24, 5]$ | $[25, 5]$ | $[24, 5]$ |
| News | $\frac{6}{50} = (12\%)$ | $\frac{44}{50} = (88\%)$ | $\frac{2}{50} = (4\%)$ | $\frac{48}{50} = (96\%)$ |
| | $[25, 0]$ | $[24, 5]$ | $[25, 5]$ | $[24, 5]$ |

Table 8: Test case, subject sport versus group news classification scores based on AMDF and OA feature vectors from four-seconds duration.

| Classes | AMDF Method | | OA Method | |
|---|---|---|---|---|
| | Sports | News | Sports | News |
| Sports | $\frac{47}{50} = (94\%)$ | $\frac{3}{50} = (6\%)$ | $\frac{49}{50} = (98\%)$ | $\frac{1}{50} = (2\%)$ |
| | $[25, 5]$ | $[24, 5]$ | $[25, 5]$ | $[24, 5]$ |
| News | $\frac{4}{50} = (12\%)$ | $\frac{46}{50} = (82\%)$ | $\frac{1}{50} = (2\%)$ | $\frac{49}{50} = (98\%)$ |
| | $[25, 5]$ | $[24, 5]$ | $[25, 5]$ | $[24, 5]$ |

Table 9: Test case, subject sport versus group news classification scores based on AMDF and OA feature vectors from five-seconds duration.

| Classes | AMDF Method | | OA Method | |
|---|---|---|---|---|
| | Sports | News | Sports | News |
| Sports | $\frac{49}{50} = (98\%)$ | $\frac{1}{50} = (2\%)$ | $\frac{50}{50} = (100\%)$ | $\frac{0}{50} = (0\%)$ |
| | $[25, 5]$ | $[24, 5]$ | $[25, 0]$ | $[25, 0]$ |
| News | $\frac{2}{50} = (4\%)$ | $\frac{48}{50} = (96\%)$ | $\frac{1}{50} = (2\%)$ | $\frac{49}{50} = (98\%)$ |
| | $[25, 5]$ | $[24, 5]$ | $[25, 0]$ | $[25, 0]$ |

Table 10: Test case, subject sport versus group news classification scores based on AMDF and OA feature vectors from six-seconds duration.

| Classes | AMDF Method | | OA Method | |
|---|---|---|---|---|
| | Sports | News | Sports | News |
| Sports | $\frac{50}{50} = (100\%)$ | $\frac{0}{50} = (0\%)$ | $\frac{50}{50} = (100\%)$ | $\frac{0}{50} = (0\%)$ |
| | $[25, 0]$ | $[25, 0]$ | $[25, 0]$ | $[25, 0]$ |
| News | $\frac{1}{50} = (2\%)$ | $\frac{49}{50} = (98\%)$ | $\frac{0}{50} = (0\%)$ | $\frac{50}{50} = (100\%)$ |
| | $[25, 0]$ | $[25, 0]$ | $[25, 0]$ | $[25, 0]$ |

These tables also present the expected frequencies accompanied by observed frequencies. Moreover, we constructed six additional tables, 11-16, which indicated the chi-square test value according to the Cramer coefficient of mean square contingency, $\phi_1$ in a 99.5 % level of significance.

Table 11: Chi-square test evaluation of the results in Table 5. Critical $\chi^2$ values from $\chi^2$ tables ($a = 0.995$ level of significance) in brackets, along with the Cramer coefficient $\phi_1$ from one-second's duration.

| **Test cases:** | **AMDF** | **OA** |
|---|---|---|
| | $\chi^2$**statistic** | $\chi^2$**statistic** |
| | $[\chi^2_{(1,0.995)}]$ | $[\chi^2_{(1,0.995)}]$ |
| | $\phi_1$ **value** | $\phi_1$ **value** |
| **Sports-News** | 54,96 | 67,3 |
| | [7.879] | [7.879] |
| | 0.27 | 0.34 |

Table 12: Chi-square test evaluation of the results in Table 6. Critical $\chi^2$ values from $\chi^2$ tables ($a = 0.995$ level of significance) in brackets, along with the Cramer coefficient $\phi_1$ from one-second's duration.

| **Test cases:** | **AMDF** | **OA** |
|---|---|---|
| | $\chi^2$**statistic** | $\chi^2$**statistic** |
| | $[\chi^2_{(1,0.995)}]$ | $[\chi^2_{(1,0.995)}]$ |
| | $\phi_1$ **value** | $\phi_1$ **value** |
| **Sports-News** | 57,76 | 81,03 |
| | [7.879] | [7.879] |
| | 0.29 | 0.41 |

For example, in Table 11, for one second the $\chi^2$ values of the test statistic, as computed from the results in Table 4, are [23.5, 26.5, 23.5, 26.5], respectively, for the AMDF vectors and [24.5, 25.5, 24.5, 25.5] for the OA vectors. As an example, for the correct positive classification cell (1,1) of Table 4 (OA vectors), $\chi^2$ test value is computed as

$\chi^2 = \frac{(42-23.5)^2}{23.5} + \frac{(8-26.5)^2}{26.5} + \frac{(5-23.5)^2}{23.5} + \frac{(45-26.5)^2}{26.5} = 54.96$

From the tables of the $\chi^2$ distribution with one degree of freedom, and at the 99.5 level of significance, we obtained the critical value 7.879, which is lower than all test statistic values. The null hypothesis of independence is therefore rejected for all six cases and for both types of feature vectors. Furthermore, the $\phi_1$ coefficient takes on values [0.27, 0.29, 0.30, 0.37, 0.44, 0.48] for the four experiments based on AMDF feature vectors and [0.34, 0.41, 0.44, 0.42, 0.48, 0.50] for the OA feature vectors. Taking into account these results we constructed figure 11.

Table 13: Chi-square test evaluation of the results in Table 7. Critical $\chi^2$ values from $\chi^2$ tables ($a = 0.995$ level of significance) in brackets, along with the Cramer coefficient $\phi_1$ from one-second's duration.

| **Test cases:** | **AMDF** | **OA** |
|---|---|---|
| | $\chi^2$**statistic** | $\chi^2$**statistic** |
| | $[\chi^2_{(1,0.995)}]$ | $[\chi^2_{(1,0.995)}]$ |
| | $\phi_1$ **value** | $\phi_1$ **value** |
| **Sports-News** | 60,86 | 88,40 |
| | [7.879] | [7.879] |
| | 0.30 | 0.44 |

Table 14: Chi-square test evaluation of the results in Table 8. Critical $\chi^2$ values from $\chi^2$ tables ($a = 0.995$ level of significance) in brackets, along with the Cramer coefficient $\phi_1$ from one-second's duration.

| **Test cases:** | **AMDF** | **OA** |
|---|---|---|
| | $\chi^2$**statistic** | $\chi^2$**statistic** |
| | $[\chi^2_{(1,0.995)}]$ | $[\chi^2_{(1,0.995)}]$ |
| | $\phi_1$ **value** | $\phi_1$ **value** |
| **Sports-News** | 73,99 | 84,80 |
| | [7.879] | [7.879] |
| | 0.37 | 0.42 |

Table 15: Chi-square test evaluation of the results in Table 9. Critical $\chi^2$ values from $\chi^2$ tables ($a = 0.995$ level of significance) in brackets, along with the Cramer coefficient $\phi_1$ from one-second's duration.

| **Test cases:** | **AMDF** | **OA** |
|---|---|---|
| | $\chi^2$**statistic** | $\chi^2$**statistic** |
| | $[\chi^2_{(1,0.995)}]$ | $[\chi^2_{(1,0.995)}]$ |
| | $\phi_1$ **value** | $\phi_1$ **value** |
| **Sports-News** | 88,83 | 96,08 |
| | [7.879] | [7.879] |
| | 0.44 | 0.48 |

Table 16: Chi-square test evaluation of the results in Table 10. Critical $\chi^2$ values from $\chi^2$ tables ($a = 0.995$ level of significance) in brackets, along with the Cramer coefficient $\phi_1$ from one-second's duration.

| **Test cases:** | **AMDF** | **OA** |
|---|---|---|
| | $\chi^2$**statistic** | $\chi^2$**statistic** |
| | $[\chi^2_{(1,0.995)}]$ | $[\chi^2_{(1,0.995)}]$ |
| | $\phi_1$ **value** | $\phi_1$ **value** |
| **Sports-News** | 96,08 | 100 |
| | [7.879] | [7.879] |
| | 0.48 | 0.5 |

As can be seen in figure 11, the superiority of the OA method compared with the AMDF method is clear in all the duration segments, and particularly in the segments between one and four seconds.

## 6 An Implementation in a Real-Time Scenario

We selected the trained LVQ neural network of five-second segmentation which yielded the best classification results. The next part was to test an unknown for the trained LVQ neural network, which belongs to the category of sports news segment, which satisfied the experimental setup settings. Thus, the original for the testing segment, consisting of $n = 22000 \cdot 5 = 111000$ values, was submitted in the OA algorithm. The result of this implementation, yielding layers $d = 842$, took place in $O(842 \cdot 111000 \log 111000) = O(1.0858e+009)$, and the feature extraction needed a time, using the Matlab 6.1 programming tool, of four seconds

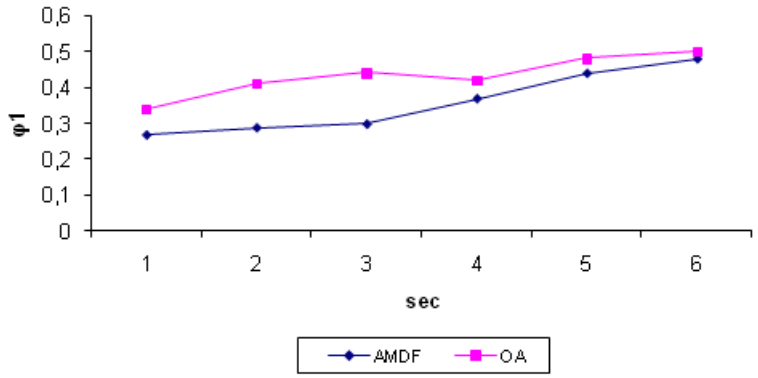


Figure 11: The ranges of $\phi_1$ coefficients between the OA (pink colour line) and the AMDF method (blue colour line).

for implementation. The testing procedure using a trained LVQ neural network needed about one millisecond. However, it must be noted that the real time is significantly lower because Matlab belongs in the interpreted languages, which are extremely slow in practice. For example, for the same algorithm the unit of measurement in Matlab is seconds, while in C it is milliseconds [2].

# 7 Discussion

The main aim of this paper has been to apply a statistical method, which we have first introduced as a pattern recognition method, to several of our own classification problems [32, 8, 6, 1, 3, 20, 31], as well as in the field of audio signal classification. This study is a consequence of our study of pattern recognition problems using computational geometric algorithms [30]. For the achievement of this aim we selected the traditional problem of audio classification files (sports and news broadcasting). More classification problems such as these are presented in the introduction. This study then showed the superiority of the OA method of feature extraction over the AMDF method by comparing their results in respect to an independent LVQ1 neural network. It then tested the ability of our proposed method using two methodological procedures: first the classification procedure and then the time learning of the input feature vectors of the LVQ1 neural network, found by measuring the minimum time error convergence which was taken as the optimal selected criterion. Moreover, as the six-seconds time segmentation for the AMDF method showed itself to be optimal in our example, this agreed with the findings of previous studies [24, 25], and thus proved the validity of our experiment. Thus, taking into account the experimental and statistical results, we may conclude that our method produces specific feature extraction coefficients which may be classified better and trained easier, with less error, than the AMDF coefficients using the LVQ1 neural network. Furthermore, we concluded that the processing time length of an audio file may be statistically accurate to greater than three seconds. In general, in future we could classify the weather report, or the political and studio news of a broadcast by adopting different philosophies of shot segmentation. Furthermore, the testing of reliability of the proposed method in relation to other philosophy neural networks such as the RBF and Recurrent classifiers is one of our upcoming objectives. The fast-learning ability of the OA coefficients of an independent neural network may be used as a feature extraction tool in more difficult audio classification problems, such as for the discrimination of a subclass of a main category of audio file, such as weather reports, political news, studio news, and so forth. The minimization of the accurate classification time length to fewer than four seconds shows that the application of our method promises to reduce the complexity significantly and to improve storage problems in pattern-recognition databases.

The results of our research show that the computational geometric algorithm is a pattern-recognition method which may be applied accurately for multimedia classification purposes. The greatest advantage of this method is that it may

be used in real time experimentation, as the feature extraction doesn't need such complex settings as the determination of interval frame, which is needed for pitch calculation. However, the achievement of this target needs further research, specifically in the reduction of its complexity, as our understanding of computational geometry improves dramatically every day. We intend the next step in our experimentation program to be in the area of larger data sets by applying this method to hierarchical classification problems involving the separation of many more categories of audio signals than the current sports and news broadcasts.

Finally, we believe that this method is the first attempt to implement the problem of the semantic classification of audio broadcasting files by using a philosophically different technique, producing a significant statistical evaluation score using Cramer criterion (see section 5). This evaluation yields a useful conclusion about the accuracy of the proposed method, giving promise that continuing research will prove useful.

## Acknowledgments

The authors wish to thank the referees for several useful suggestions.

# References

[1] http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm.

[2] http://www.mines.utah.edu/geo/facilities/computing/tutorials/unix/node4.html.

[3] B. Arons. SpeechSkimmer: a system for interactively skimming recorded speech. In *ACM trans on computer-human interaction*, volume 4, pages 3–38, 1997.

[4] P. Bose and G. Toussaint. No quadrangulation is extremely odd. In *Sixth international symposium on algorithms and computation (formerly SIGAL International Symposium on Algorithms)*, pages 340–358, 1995.

[5] K. Dalal. Counting the onion. *Random Struct Algorithms*, 24:155–165, 2001.

[6] J. Deller, M. Ayer, and S. Odeh. Least square identification with error bounds for real-time signal processing and control. In *Proc IEEE 93*, volume 81, pages 813–849, 1993.

[7] R. Graham. An efficient algorithm for determining the convex hull of a finite planar set. *Information process, Lett*, 1:132–133, 1972.

[8] A. Hauptmann and M. Witbrock. Informedia: news-on-demand multimedia information acquisition and retrieval. In *Intelligent multimedia information retrieval*, pages 215–240. MIT Press, Cambridge, 1997.

[9] S. Haykin. *Neural Networks*. MacMillan, New York, 1994.

[10] W. Hess. *Pitch Determination of Speech Signals*. Springer Verlag, Heidelberg, 1983.

[11] A. Hoekstra and R. Duin. Exploring the capacity of simple neural networks. In J. Katwijk, J. Gerbrands, van M. Steen, and J. Tonino, editors, *Proceedings of the first annual conference of the ASCI*, pages 56–62, 1995.

[12] A. Jain and W. Waller. On the optimal number of features in the classification of multivariate gaussian data. *Pattern recognition*, 10:365–374, 1978.

[13] N. Kanal and B. Chandrasekaran. On dimensionality and sample size in statistical pattern recognition. *Pattern recognition*, 3:225–234, 1971.

[14] J. Kangas, T. Kohonen, and J. Laakson. Variants of self-organizing maps. *IEEE trans neural networks*, 1:93–99, 1990.

[15] K.Torkkola, J.Kangas, P. Utela, S. Kaski, M. Kokkonen, M. Kurimo, and T. Kohonen. Status report of the finnish phonetic typewriter project. ICANN91, 1991.

[16] Z. Liu, J. Huang, Y. Wang, and T. Chen. Audio feature extraction and analysis for scene classification. In *Workshop on Multimedia Signal Processing*, volume 20, pages 1–2. IEEE 97 Signal Processing Society, 1997.

[17] L. Lu, H. Jiang, and J. Zhang. A robust audio classification and segmentation method. In *Proc. of the 9th ACM int conf on multimedia*, pages 203–211, 2001.

[18] H. Man and et. al. Three-dimensional sub-band coding techniques for wireless video communications. In *IEEE trans on circuits and systems for video tech*, volume 12, pages 386–397, 2002.

[19] J. O' Rourke. *Computational Geometry in C*. Spencer T. Cambridge University Press, New York, 1993.

[20] M. Orlandi, A. Santarelli, and D. Falavigna. Maximum likelihood endpoint detection with time-domain features. In *Proc of eurospeech*, pages 1757–1760, 2003.

[21] A. Orlitsky. Supervised dimensionality reduction using mixture models. In *Twenty-second int conf on machine learning*, pages 768–775, 2005.

[22] M. Poulos, N. Alexandris, V. Belessioti, and E. Magkos. Comparison between computational geometry and coherence methods applied to the EEG for medical diagnostic purposes. In *Recent Advances in intelligent Systems and Signal Processing, ICAISC*, 2003.

[23] M. Poulos, N. Alexandris, V. Belessioti, and E. Magkos. Computational geometry algorithms in an educational intelligent scenario management system. In N. Mastorakis, N. Manikopoulos, C. Antonioy, and V. Maladenov, editors, *Recent Advances in Intelligent Systems and Signal Processing, ICAISC*, pages 242–246, 2003.

[24] M. Poulos, N. Korfiatis, and S. Papavlasopoulos. Anti-spam filtering using computational geometry. *WSEAS transactions on information science & applications*, pages 747–751, 2004.

[25] M. Poulos, S. Papavlasopoulos, and V. Chrissicopoulos. A text categorization technique based on a numerical conversion of a symbolic expression and an onion layers algorithm. *Journ of Digital Inf (JoDI)*, 6(1):176, 2004.

[26] M. Poulos, S. Papavlasopoulos, V. Chrissicopoulos, and E. Magkos. Fingerprint verification based on image processing segmentation using an onion algorithm of computational geometry. In *In Sixth Int Conf on Mathematics Methods in Scattering Theory and Biomedical Technology*, pages 550–559. BIOTECH 03, Word Scientific, Tsepelovo-Ioannina, 2003.

[27] M. Poulos, M. Rangous, and E. Kafetzopoulos. Person identification via the EEG using computational geometry algorithms. In S. Theodoridis, A. P. N. Stouraitis, and N. Kalouptsidis, editors, *Proceedings of the Ninth European Signal Processing*, volume 4, pages 2125–2212. EUSIPCO 98, Rhodes, 1998.

[28] M. Poulos, M. Rangousi, N. Alexandris, and A. Evangelou. Person identification from the EEG using nonlinear signal classification. *Methods of info in medicine*, 41:64–74, 2001.

[29] M. Poulos, M. Rangoussi, V. Chrissicopoulos, and A. Evangelou. Parametric person identification from the EEG using computational geometry. In *IEEE proceedings of the 6th Int conf on electronics circuits and systems*, volume 2, pages 1005–1012. ICECS 99, inst of electrical and electronics engineers, 1999.

[30] B. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, 1996.

[31] M. J. Ross, H. L. Shaffer, A. Cohen, R. Freudberg, and H. J. Manley. Average magnitude difference function pitch extractor. In *IEEE trans on acoustics, speech, and signal processing*, volume 22, pages 353–362, 1974.

[32] J. Saunders. Real-time discrimination of broadcast speech/music. In *ICASSP96*, volume 2, pages 993–996, 1996.

[33] E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature music/speech discriminator. In *Proc of ICASSP 97*, volume 2, pages 1331–1334, 1997.

[34] S. Smoliar and H. Zhang. Content-based video indexing and retrieval. *IEEE multimedia mag,*, 1:62–72, 1994.

[35] G. Tzanetakis and M. Chen. Building audio classifiers for broadcast news retrieval. In *Proc. of WIAMIS 04, Portugal*. WIAMIS 04, Portugal, 2004.

[36] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. In *IEEE 02 Trans on speech and audio proc*, volume 10, pages 293–302, 2002.

[37] J. Zar. *Biostatistical Analysis (4th Edition)*. Prentice-Hall, London, 1999.