Audio Engineering Society
Convention Paper
Presented at the 115th Convention
2003 October 10–13 New York, NY, USA

Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization

Geoffroy Peeters[1]

[1] *IRCAM, 1 pl. Igor Stravinsky, 75004 Paris, France*

Correspondence should be addressed to Geoffroy Peeters (peeters@ircam.fr)

ABSTRACT

This paper addresses the problem of classifying large databases of musical instrument sounds. An efficient algorithm is proposed for selecting the most appropriate signal features for a given classification task. This algorithm, called IRMFSP, is based on the maximization of the ratio of the between-class inertia to the total inertia combined with a step-wise feature space orthogonalization. Several classifiers - flat gaussian, flat KNN, hierarchical gaussian, hierarchical KNN and decision tree classifiers - are compared for the task of large database classification. Especially considered is the application when our classification system is trained on a given database and used for the classification of another database possibly recorded in completely different conditions. The highest recognition rates are obtained when the hierarchical gaussian and KNN classifiers are used. Organization of the instrument classes is studied through an MDS analysis derived from the acoustic features of the sounds.

1. INTRODUCTION

During the last decades, sound classification has been the subject of many research efforts [27] [3][17] [29]. However, few of them address the problem of generalization of the sound source recognition system i.e. applicability to several instances of the same source possibly recorded in different conditions, with various instrument manufacturers and players. In this context, Martin [16] reports only 39% recognition rate for individual instrument (76% for instrument family), using the output of a log-lag correlogram for 14 different instruments. Eronen [4] reports 35% (77%) recognition rate using mainly MFCCs and some other features for 16 different instruments.

Sound classification systems rely on the extraction of a set of signal features (such as energy, spectral centroid, ...) from the signal. This set is then used to perform classification according to a given taxonomy. This taxonomy is defined by a set of textual attributes defining the properties of the sound such as its source (speaker genre, music genre, sound effects class, instrument name...) or its perception (bright, dark...).

The choice of the features depends on the targeted application (speech/music/noise discrimination, speaker identification, sound effects recognition, musical instruments recognition). The most appropriate set of features can be selected a priori - having a prior knowledge of the feature discriminative power for the given task -, or a posteriori by including in the system an algorithm for automatic feature selection. In our system, in order to allow the coverage of a large set of potential taxonomies, we have implemented a large set of features. This set of features is then filtered automatically by a feature selection algorithm.

Because sound is a phenomenon, which changes over time, features are computed over time (frame by frame analysis). The set of temporal features can be used directly for classification [3]; or the temporal evolution of the features can be modeled. Modeling can be done before the modeling of the classes (using mean, std, derivative values, modulation or polynomial representation [29]) or during the modeling of the classes (using for example a Hidden Markov Model [30]). In our system, temporal modeling is done before that of the classes.

The last major difference between classification systems concerns the choice of the model to represent the classes of the taxonomy (multi-dimensional gaussian, gaussian mixture, KNN, NN, decision tree, SVM...).

The system performance is generally evaluated, after training on a subset of a database, on the rest of the database. However, since most of the time a single database contains a single instance of an instrument (the same instrument played by the same player in the same recording conditions), this kind of evaluation does not prove any applicability of the system for the classification of sounds which do not belong to the database. In particular, the system may fail to recognize sounds recorded in completely different conditions. In this paper we evaluate such performances.

2. FEATURE EXTRACTION

Many different types of signal features have been proposed for the task of sound recognition coming from the speech recognition community, previous studies on musical instrument sounds classification [27] [3] [17] [29] [13] and results of psycho-acoustical studies [14] [24]. In order to allow the coverage of a large set of potential taxonomies, a large set of features has been implemented, including features related to the

- *Temporal shape* of the signal (attack-time, temporal increase/decrease, effective duration),
- *Harmonic features* (harmonic/noise ratio, odd to even and tristimulus harmonic energy ratio, harmonic deviation),
- *Spectral shape features* (centroid, spread, skewness, kurtosis, slope, roll-off frequency, variation),
- *Perceptual features* (relative specific loudness, sharpness, spread, roughness, fluctuation strength),
- Mel-Frequency Cepstral Coefficients (plus Delta and DeltaDelta coefficients), auto-correlation coefficients, zero-crossing rate, as well as some MPEG-7 Low Level Audio Descriptors (spectral flatness and crest factors [22]).

See [12] for a review.

3. FEATURE SELECTION

Using a high number of features for classification can cause several problems: 1) bad classification results because some features are irrelevant for the given task; 2) over fitting of the model to the training set (this is especially true when using, without care, data reduction techniques such as Linear Discriminant Analysis), 3) the models are difficult to interpret by human. For this reason, feature selection algorithms attempt to detect the minimal set of

1. informative features with respect to the classes
2. features that provide non redundant information.

3.1. Inertia Ratio Maximization using Feature Space Projection (IRMFSP)

Feature selection algorithms (FSA) can take three main forms (see [21]):

- embedded: the FSA is part of the classifier
- filter: the FSA is distinct from the classifier and used before the classifier
- wrapper: the FSA makes use of the classification results.

The FSA we propose is part of the Filter techniques.

Considering a gaussian classifier, the *first criterion* for FSA can be expressed in the following way: "feature values for sounds belonging to a specific class should be separated from the values for all the other classes". If it is not the case then the gaussian pdfs will overlap, and class confusion will increase. In a mathematical way this can be expressed by looking at features for which the ratio r of the Between-class inertia B to the Total class inertia T is maximum. For a specific feature f_i, r is defined as

$$r = \frac{B}{T} = \frac{\sum_{k=1}^{K} \frac{N_k}{N} (m_{i,k} - m_i)(m_{i,k} - m_i)'}{\frac{1}{N} \sum_{n=1}^{N} (f_{i,n} - m_i)(f_{i,n} - m_i)'} \quad (1)$$

where N is the total number of data, N_k is the number of data belonging to class k, m_i is the center of gravity of the feature f_i over all the data set, and $m_{i,k}$ is the center of gravity of the feature f_i for data belonging to class k. A feature f_i with a high value of r is therefore a feature for which the classes are well separated with respect to their within spread.

The *second criterion* should allow taking into account the fact that a feature with a high value of r could bring the same information as an already selected feature and is therefore redundant. While other FSAs, like the CFS one [10][1], use a weight based on the correlation between the candidate feature and already selected features, in the IRMFSP algorithm, an orthogonalization process is applied after the selection of each new feature f_i. If we note $\underline{\underline{F}}$ the feature space (space where each axis represents a feature), $\underline{f}_i$ the last selected feature and $\underline{g}_i$ its normalized form ($\underline{g}_i = \underline{f}_i / ||\underline{f}_i||$), we project $\underline{\underline{F}}$ on $\underline{g}_i$and keep $\underline{f}'_j$:

$$\underline{f}'_j = \underline{f}_j - \left(\underline{f}_j \cdot \underline{g}_i\right) \underline{g}_i \; \forall j \in \underline{\underline{F}} \quad (2)$$

This process (ratio maximization followed by space projection) is repeated until the gain of adding a new feature $\underline{f}_i$ is too small. This gain is measured by the ratio r_l obtained at the l^{th} iteration to the one at the first iteration. A stopping criterion of $t = \frac{r_l}{r_1} < 0.01$ has been chosen.

In Fig.1, we illustrate the results of the IRMFSP algorithm for the selection of features for a two classes taxonomy: separation between sustained and non-sustained sounds. In Fig.1, sounds are represented along the first three selected dimensions: temporal decrease (1st dim), spectral centroid (2nd dim) and temporal increase (3rd dim).

In part 6.2, the CFS and IRMFSP algorithm are compared.

4. FEATURE TRANSFORMATION

In the following, two feature transformation algorithms (FTA) are considered.

[1] In the CFS algorithm (Correlation-based Feature Selection), the information brought by one specific feature is computed using symmetrical uncertainty (normalized mutual information) between discretized features and classes. The second criterion (features independence) is taken into account by selecting a new feature only if its cumulated correlation with already selected features is not too large.

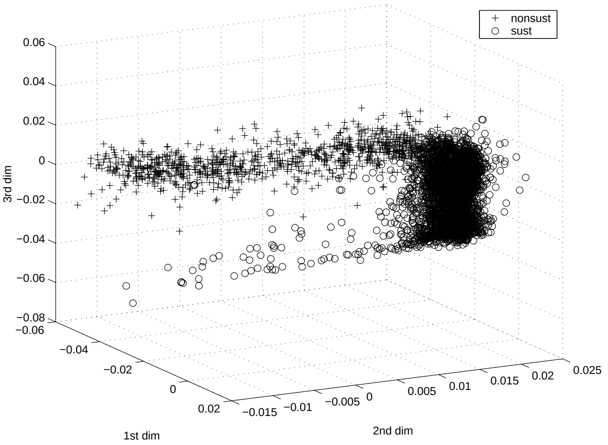

Fig. 1: First three dimensions selected by the IRMFSP algorithm for the sustained / non-sustained sounds taxonomy

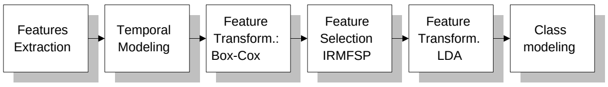

Fig. 2: Classification system flowchart

4.1. Box-Cox Transformation

Classification models based on gaussian distribution makes the underlying assumption that modeled data (in our case signal features) follow a gaussian probability density function (pdf). However, this is rarely verified by features extracted from the signal. Therefore a *first* FTA, a non-linear transformation, known as the “Box-Cox transformation” [2], can be applied to each feature individually in order to make its pdf fit as much as possible a gaussian pdf. The set of considered non-linear functions depending on the parameters λ is defined as

$$\begin{aligned} f_\lambda(x) &= \frac{x^\lambda - 1}{\lambda} \text{ if} \lambda \neq 0 \\ f_\lambda(x) &= \log(x) \text{ if} \lambda = 0 \end{aligned} \quad (3)$$

For a specific value of λ, the gaussianity of $f_\lambda(x)$ is measured by the correlation factor between the percent point function ppf (inverse of the cumulative distribution) of $f_\lambda(x)$ and the theoretical ppf of a gaussian function. For each feature x, we find the best non-linear function (best value of λ) defined as the one with the largest gaussianity.

4.2. Linear Discriminant Analysis

The *second* FTA is the Linear Discriminant Analysis (LDA) which was proposed by [17] in the context of musical instrument sound classification and evaluated successfully in our previous classifier [25]. LDA allows finding a linear combination among features in order to maximize discrimination between classes. From the initial feature space $\underline{\underline{F}}$ (or a selected feature space $\underline{\underline{F}}'$), a new feature space $\underline{\underline{G}}$ of dimension smaller than $\underline{\underline{F}}$ is obtained.

In our current classification system, LDA (when performed) is used between the feature selection algorithm and the class modeling (see Fig.2).

5. CLASS MODELING

Among the various existing classifiers (multi-dimensional gaussian, gaussian mixture, KNN, NN, decision-tree, SVM...) (see [12] for a review), only the gaussian, KNN (and their hierarchical formulation) and decision-tree classifiers have been considered.

5.1. Flat Classifiers

5.1.1. Flat gaussian classifier (F-GC)

A flat gaussian classifier models each class k by a multi-dimensional gaussian pdf. The parameters of the pdf (mean μ_k and covariance matrix $\underline{\underline{\Sigma}}_k$) are estimated by maximum-likelihood given the selected features for sounds belonging to class k. The term ”flat” is used here since all classes are considered on a same level. In order to evaluate the probability that a new sound belongs to a class k, Bayes formula is used:

$$p(k|\underline{f}) = \frac{p(\underline{f}|k)p(k)}{p(\underline{f})} = \frac{p(\underline{f}|k)p(k)}{\sum_k (p(\underline{f}|k)p(k)} \quad (4)$$

where - $p(k)$ is the prior probability of observing class k, - $p(\underline{f})$ is the distribution of the feature-vector $\underline{f}$ - $p(\underline{f}|k)$ is the conditional probability of observing the feature-vector given a class k (the estimated gaussian pdf).

The training and evaluation process of a flat gaussian classifier system is illustrated in Fig.3.

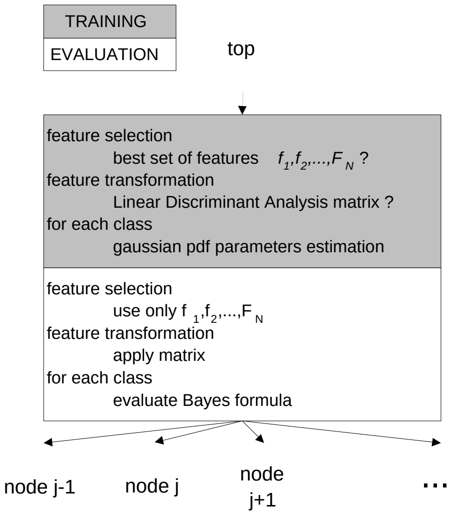

Fig. 3: Flat gaussian classifier

5.1.2. Flat KNN classifiers (F-KNN)

K Nearest Neighbors (KNN) is one of the most straightforward algorithm for data classification. KNN is an instance-based algorithm. In KNN, the position of the data of the training set in the feature space $\underline{\underline{F}}$ (or in the selected feature space $\underline{\underline{F'}}$) is simply stored (without modeling) along with the corresponding classes. For an input sound located in $\underline{\underline{F}}$, the K closest data of the training set (the K Nearest Neighbors) are estimated. The majority class among these KNN is assigned to the input sound. An Euclidean distance is commonly used in order to find the K closest data. Therefore the weighting of the axes of the space $\underline{\underline{F}}$ (weighting of the features) can change the closest data. In the following of this study, when using KNN classifiers, the weighting of the axes is implicitly done since the KNN is applied to the output space of the LDA transformation (LDA finds the optimal weights for the axes of the feature space $\underline{\underline{G}}$). The number of considered nearest neighbors, K, also plays an important role in the obtained result. In the following of this study, the results are indicated for a value of K=10 which yields to the best results in our case.

5.2. Hierarchical Classifiers

5.2.1. Hierarchical gaussian classifier (H-GC)

A hierarchical gaussian classifier is a tree of flat gaussian classifiers, i.e. each node of the tree is a flat gaussian classifier with its own feature selection (IRMFSP), its own LDA, its own gaussian pdfs. Hierarchical classifiers have been used by [17] for the classification of 14 instruments (derived from the McGill Sound Library) using a hierarchical KNN-classifier and Fisher multiple discriminant analysis combined with a gaussian classifier. During the training, only the subset of sounds belonging to the classes of the current node (example: the bowed-string node is trained using only bowed-string sounds, the brass node is trained using only brass sounds) is used. During the evaluation, the maximum local probability at each node (probability $p(k|\underline{f})$) decides which branch of the tree to follow. The process is then pursued until reaching a leaf of the tree.

Contrary to binary trees, the construction of the tree structure of a H-GC is supervised and requires a previous knowledge of class organization (oboe belongs to double-reeds family which belongs to sustained sounds).

Advantages of Hierarchical Gaussian Classifiers (H-GC) over Flat Gaussian Classifiers (F-GC) .

Learning facilities: Learning a H-GC (feature selection and gaussian pdf model parameter estimation) is easier since it is easier to characterize the difference in a small subset of classes (learning the difference between brass instruments only is easier than between the whole set of classes).

Reduced class confusion: In a F-GC, all classes are represented on the same level and are thus neighbors in the same multi-dimensional feature space. Therefore, annoying class confusions, as for example confusing an "oboe" sound with an "harp" sound, are likely to occur. In a H-GC, because of the hierarchy and the high recognition rate at the higher levels of the tree (such as non-sustained /sustained sounds node), this kind of confusion is unlikely to occur.

The training and evaluation process of a hierarchical gaussian classifier system is illustrated in Fig.4. The gray/white box connected to each node of the tree is the same as the one of Fig.3.

5.2.2. Hierarchical KNN classifiers (H-KNN)

In hierarchical KNN, at each level of the tree, only the locally selected features and the locally considered classes are taken into account for the training.

The training and evaluation process of a hierarchical KNN classifier system is illustrated in Fig.4.

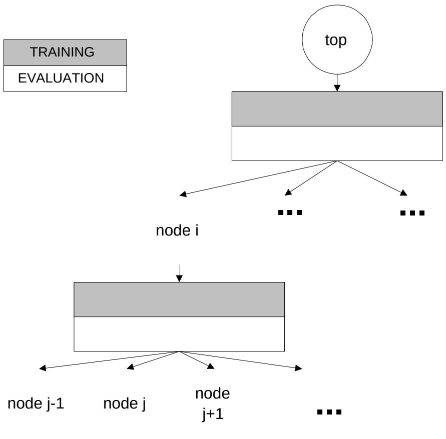

Fig. 4: Hierarchical classifier

5.3. Decision Tree Classifiers

Decision trees operate by asking the data a sequence of questions in which the next question depends on the answer to the current question. Since they are based on questions they can operate on both numerical and non-numerical data.

5.3.1. Binary Entropy Reduction Tree (BERT)

A binary tree recursively decomposes the set of data into two subsets of data in order to maximize one class belonging. The decomposition is operated by a split criterion. In the binary tree considered here, the split criterion operates on a single variable. The split criterion decides with respect to the feature value which branch of the tree to follow (if feature $<$ threshold then left branch, if feature $\geq$ threshold then right branch). In order to automatically determine the best feature and the best value of the threshold at each node, mutual information and binary entropy are used as in [5].

The mutual information between a feature X, for a threshold t and classes C can be expressed as

$$I(X,C) = H_2(X|t) - H_2(X|C,t) \tag{5}$$

where $H_2(X|C,t)$ is the binary entropy given the classes C and given the threshold t:

$$H_2(X|C,t) = \sum_k p(C_k) H_2(X|C_k,t) \tag{6}$$

and $H_2(X|t)$ is the binary entropy given the threshold t:

$$H_2(X|t) = -p(x)\log_2(p(x)) - (1-p(x))\log_2(1-p(x)) \tag{7}$$

where p is the probability that $x < t$, $(1-p)$ that $x \geq t$

The best feature and the best threshold value at each node are the ones for which I(X,C) is maximum.

Pre-pruning of the tree: The tree construction is stopped when the gain of adding a new split is too small. The stopping criterion used in [5] is the mutual information weighted by the local mass inside the current node j:

$$\frac{N_j}{N} I_j(X,C) \tag{8}$$

In part 6.2, the results obtained with our Binary Entropy Reduction Tree (BERT) and with two other widely used decision tree algorithms: C4.5. [26] and Partial Decision Tree (PART) [7] are compared.

6. EVALUATION

6.1. Methodology

6.1.1. Evaluation process

For the evaluation of the models, three methods have been used.

The *first evaluation* method used is the random 66%/33% partition of the database where 66% of the sounds of each class of a database are randomly selected in order to train the system. The evaluation is then performed on the remaining 33%. In this case, the result is given as the mean value over 50 random sets.

The second and third evaluation methods were proposed by Livshin [15] for the evaluation of large database classification, especially for testing the applicability of a system trained on a given database when used for the recognition of another database

The *second evaluation* method, called O2O (One to One), uses in turns each database for training the system and measure the recognition rate on each of the remaining ones. If we note A, B and C the various databases, the training is performed on A, and used for the evaluation of B and C; then the training is performed on B, and used for the evaluation of A and C, ...

The *third evaluation* method, called the LODO (Leave One Database Out), uses all databases for the training except one which is used for the evaluation. All possible left out databases are chosen in turns. The training is performed on A+B, and used for the evaluation of C; then the training is performed on A+C, and used for the evaluation of B; ...

6.1.2. Taxonomy used

The instrument taxonomy used during the experiment is represented in Fig.5. In the following experiments we consider taxonomies at three different levels:

1. a 2 classes taxonomy: sustained/ non-sustained sounds. We call it T1 in the following.
2. a 7 classes taxonomy corresponding to the instrument families: struck strings, plucked-strings, pizzicato-strings, bowed-strings, brass, air reeds, single/double reeds. We call it T2 in the following.
3. a 27 classes taxonomy corresponding to the instrument names: piano, guitar/ harp, pizzicato-violin/ viola/ cello/double-bass, bowed-violin/ viola/ cello/ double-bass, trumpet/ cornet/ trombone/ FrenchHorn/ tubba, flute/ piccolo/ recorder, oboe/ bassoon/ EnglishHorn/ clarinet/ accordion/ alto-sax/ soprano-sax/ tenor-sax. We call it T3 in the following.

This taxonomy is of course subject to discussions, especially - the piano, which is supposed to belong here to the non-sustained family - the inclusion of all saxophone instruments in the same family as the oboe.

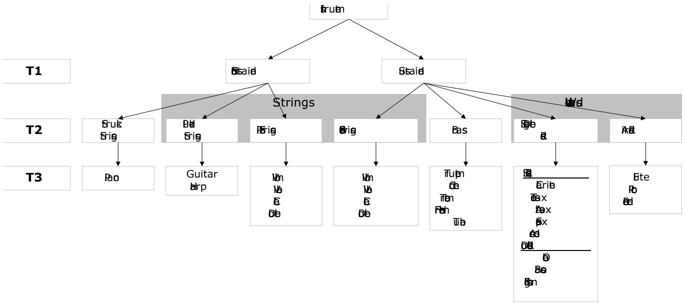

Fig. 5: Instrument Taxonomy used for the experiment

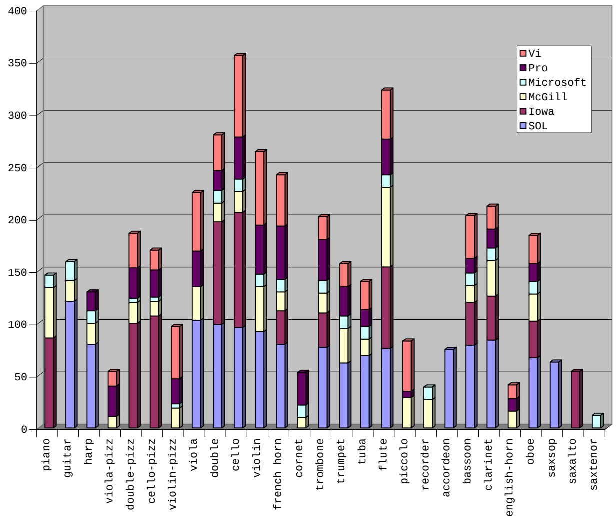

Fig. 6: Instrument distribution of the six database

6.1.3. Test set

Six different databases were used for the evaluation of the models:

- the Ircam Studio OnLine [1] (1323 sounds, 16 instruments),

- the Iowa University database [8] (816 sounds, 12 instruments),
- the McGill University database [[23] (585 sounds, 23 instruments),
- sounds extracted from the Microsoft ”Musical Instruments” CD-ROM [19] (216 sounds, 20 instruments),
- two commercial databases the Pro (532 sounds, 20 instruments) and the Vi databases (691 sounds, 18 instruments),for a total of 4163 sounds.

It is important to note that a large pitch range has been considered for each instrument (4 octaves on average). In the opposite, not all the sounds from each database have been considered. In order to limit the number of classes, the muted sounds, the martele/ staccato sounds and some more specific type of playing have not considered been. The instrument distribution of each database is depicted in Fig.6.

6.2. Results

6.2.1. Comparison of feature selection algorithms

In Table 1, we compare the result of our previous classification system [25] (which was based on Linear Discriminant Analysis applied to the whole set of features combined with a flat gaussian classifier) with the results obtained with the flat gaussian classifier applied directly (without feature transformation) to the output of the two feature selection algorithms CFS and IRMFSP. The result is given for the Studio OnLine database for taxonomies T1, T2 and T3. Evaluation is performed using the 66%/33% paradigm with 50 random sets.

Discussion: Comparing the result obtained with our previous classifiers (LDA) and the result obtained with the IRMFSP algorithm, we see that using a good feature selection algorithm not only allows to reduce the number of features but also increases the recognition rate. Comparing the results obtained using the CFS and IRMFSP algorithms, we see that for T3 IRMFSP performs better than CFS. Since the number of classes is larger at T3, the number of required features is also larger and features redundancy is more likely to occur. CFS fails at T3, perhaps because of a potentially high feature redundancy.

Table 1: Comparison of feature selection algorithm in terms of recognition rate, mean (standard deviation)

	T1	T2	T3
LDA	96	89	86
CFS weka	99.0 (0.5)	93.2 (0.8)	60.8 (12.9)
IRMFSP (t=0.01, nbdesc-max=20)	99.2 (0.4)	95.8 (1.2)	95.1 (1.2)

6.2.2. Comparison of classification algorithms for cross-database classification

In Table 2, we compare the recognition rate obtained using the O2O evaluation method for the

- Flat gaussian (F-GC) and
- Hierarchical gaussian (H-GC) classifiers.

The results are indicated as mean values over the 30 (6*5) O2O experiments (six databases). Feature transformation algorithms (Box-Cox and LDA transformations) are not used here considering that the number of data inside each database is too small for a correct estimation of FTA parameters. Features have been selected using the IRMFSP algorithm with a stopping criterion of t$\leq$0.01 and a maximum of 10 features per node.

Discussion: Compared to the results of Table 1, we see that good result with flat gaussian classifier using 66%/33% paradigm on a single database does not prove any applicability of the system for the recognition of another database (30% using F-GC at T3 level). This is partly explained by the fact that each database contains a single instance of an instrument (same instrument played by the same player in the same recording conditions). Therefore the system mainly learns the instance of the instrument instead of the instrument itself and is unable to recognize another instance of it. Results obtained using H-GC are higher than with H-GC (38% at T3 level).

Table 2: Comparison of flat and hierarchical gaussian classifiers using O2O methodology

	T1	T2	T3
F-GC	89	57	30
H-GC	93	63	38

This can be partly explained by the fact that, in a H-GC, lower levels of the tree benefit from the classification results of higher levels. Since the number of instances used for the training at the higher level is larger (at the T2 level, each family is composed of several instruments, thus several instances of the family) the training of higher level can be generalized and the lower level benefits from this.

Not indicated here are the various recognition rates of each individual O2O experiment. These results show that when the training is performed on either Vi, McGill or Pro database, the model is applicable for the recognition of most other databases. On the other hand, when training is performed on Iowa database, the model is poorly applicable to other databases.

6.2.3. Comparison of classification algorithms for large database classification

In order to increase the number of possible instrument models, several databases can be combined as in the LODO evaluation method.

In Table 3, we compare the recognition rate obtained using the LODO evaluation method for the

- Flat classifiers: flat gaussian (F-GC) and flat KNN (F-KNN)
- Hierarchical classifiers: hierarchical gaussian (H-GC) and hierarchical KNN (H-KNN)
- Decision tree classifiers: Binary Entropy Reduction Tree (BERT), C4.5. and PART.

The results are indicated as mean values over the six Left Out databases. For flat and hierarchical classifiers (F-GC, F-KNN, H-GC and H-KNN), features have been selected using the IRMFSP algorithm with a stopping criterion of t≤0.01 and a maximum of 40 features per node. For F-KNN and H-KNN, LDA has been applied at each node in order to maximize class separation and to obtain the proper weighting of the KNN axes.

Comparing O2O and LODO results: As expected, the recognition rate increases with the number of instances of each instrument used for the training (for F-GC at T3 level: 30% using O2O and 53% using LODO, for H-GC at T3 level: 38% using O2O and 57% using LODO).

Comparing flat and hierarchical classifiers: The best results are obtained with the hierarchical classifiers, both H-GC and H-KNN. In Table 3, the effect of applying feature transformation algorithm (Box-Cox and LDA transformations) for both F-GC and H-GC is observed. In the case of H-GC, it increases the recognition rate from 57% to 64%. It is commonly held that among classifiers, KNN provides the highest recognition rates. However in our case, H-KNN and H-GC (when combined with feature transform) provide very similar results: H-KNN: T1=99%, T2=84%, T3=64% and H-GC: T1=99%, T2=85%, T3=64%.

Decision Tree algorithm: Using decision tree classifiers surprisingly yields poor results even when using post-pruning techniques (such as the ones of C4.5). This is surprising considering the high recognition rate obtained by [11] for the task of unpitched percussion sounds recognition. This tends to favor the use of "smooth" classifiers (based on probability) instead of "hard" classifiers (based on Boolean boundaries) for the task of musical instrument sounds recognition. Among the various tested decision tree classifiers, the best results were obtained using Partial Decision Tree algorithm for the T2 level (T2=71%) and C4.5 algorithm for the T3 level (T3=48%).

6.3. Instrument Class Similarity

For the learning of hierarchical classifiers, the construction of the tree structure is supervised and based on a prior knowledge of class proximity (for example violin is close to viola but far from piano). It is therefore interesting to verify whether the assumed structure used during the experiment (see Fig.5) corresponds to a "natural" organization of sound classes. In order to check the assumed structure, several possibilities can be considered as the analysis of the class distribution among the leaves of a decision tree.

Table 3: Comparison of flat, hierarchical and decision tree classifiers using LODO methodology

	T1	T2	T3
F-GC	98	78	55
F-GC (BC+LDA)	99	81	54
F-KNN (K=10, LDA)	99	77	51
H-GC	98	80	57
H-GC (BC+LDA)	99	85	64
H-KNN (K=10, LDA)	99	84	64
BERT	95	65	42
C4.5.		65	48
PART		71	42

Herrera proposed in [11] an interesting method in order to estimate similarities and differences between instrument classes in the case of unpitched percussion sounds. This method allows the estimation of a two-dimensional map obtained by Multi-Dimensional scaling analysis of a similarity matrix between class parameters. Multi-dimensional Scaling (MDS) allows representing a set of data observed through their dissimilarities into a low-dimensional space such that, in this space the distances between the data is preserved as much as possible. MDS has been used to represent the underlying perceptual dimension of musical instrument sounds in a low-dimensional space [20] [9] [18]. In these studies, people were asked for dissimilarity judgements on pairs of sounds. MDS was then used to represent the stimuli into a lower dimensional space. In [18], a three-dimensional space has been found for musical instrument sounds with the three axes assigned to the attack time, the brightness and the spectral flux of sounds.

In [11], the MDS representation is derived from the acoustic features (signal features) instead of dissimilarity judgements. A similar approach is followed here for the case of musical instrument sounds.

Our classification system has been trained using a flat gaussian classifier (without any assumption related to classes proximity) and the whole set of databases. Resulting from this training is the representation of each instrument class in terms of acoustic parameters (mean vector and covariance matrix for each class). The "between-groups F-matrix" is computed from the class parameters and used as an index of similarity between classes. An MDS analysis (using Kruskal's STRESS formula 1 scaling method) is then performed on this similarity matrix. The results from the MDS analysis is a three-dimensional space represented in Fig.7. The instrument name abbreviations used in Fig.7 are explained in Table 6.3. Since this low-dimensional space is supposed to preserve (as much as possible) the similarity between the various instrument classes, it should allow identifying possible class organization.

Dimension 1 separates the non-sustained sounds on the negative values (PIAN, GUI, HARP, VLNP, VLAP, CELLP, DBLP) from the sustained sounds on the positive values. Dimension 1 seems therefore to be associated to both the attack-time and decrease time. Dimension 2 could be associated to brightness since it separates some dark sounds (TUBB, BSN, TBTB, FHOR) from some bright sounds (PICC, CLA, FLTU) although some sounds such as the DBL contradicts this assumption. Dimension 3 remains unexplained except that it allows the separation of bowed-strings (VLN, VLA, CELL, DBL) from the other instruments and that it could therefore be explained by the amount of modulation of the sounds.

In Fig.7, several clusters are observed: the bowed-string sounds (VLN, CLA, CELL, DBL), the brass sounds (TBTB, FHOR, TUBB with the exception of TRPU) and the non-sustained sounds (PIAN, GUI, HARP, VLNP, VLAP, CELLP, DBLP). Another cluster appears in the center of the space containing a mix between single/double reeds and brass instruments (SAXSO, SAXAL, SAXTE, ACC, EHOR, CORN).

From this analysis, it appears that the assumed class structure is only partly verified by the analysis of the MDS map. Only the non-sustained brass and bowed-string families are observed as clusters in the MDS map.

7. CONCLUSION

In this paper we investigated the classification of large musical instrument databases. We proposed a new feature selection algorithm based on the maximization of the ratio of the between-class inertia to the total inertia, and compared it successfully with the widely used CFS algorithm. We compared various classifiers: gaussian, KNN classifiers,

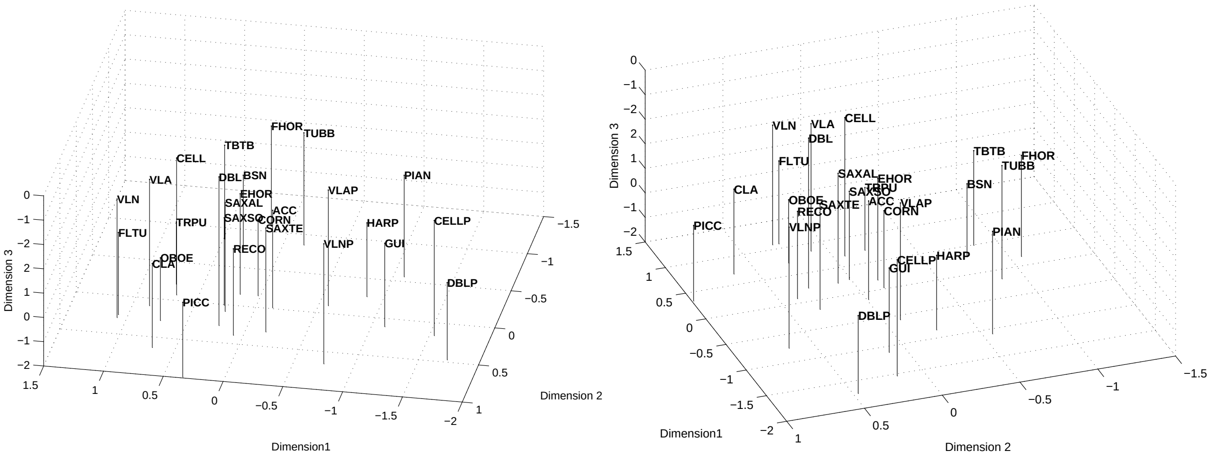

Fig. 7: Multi-dimensional scaling solution for musical instrument sounds: two different angles of view of the 3-dimensional map

their corresponding hierarchical form and various decision tree algorithms. The highest recognition rates were obtained when hierarchical gaussian and KNN classifiers are used. This tends to favor the use of “smooth” classifiers (based on probability like the gaussian classifier) instead of “hard” classifiers (based on Boolean boundaries like the decision tree classifier) for the task of musical instrument sounds recognition. In order to validate the class hierarchy used in the experiment, we studied the organization of the classes through an MDS analysis using an acoustic feature representation of the instrument classes. This study leads to the conclusion that non-sustained bowed-string and brass instrument families form clusters in the acoustic feature space, while the rest of the instrument families (reed families) are at best sparsely grouped. This is also verified by the analysis of the confusion matrix.

The recognition rate obtained with our system (64% for 23 instruments, 85% for instrument families) must be compared to the results reported by previous studies: Martin (respectively Eronen), 39% for 14 instruments, 76% for instrument families (respectively 35% for 16 instruments, 77% for instrument families). The increased recognition rates obtained in the present study can be mainly attributed to the use of new signal features.

APPENDIX

In Fig.8, we present the main selected features by the IRMFSP algorithm at each node of the H-GC tree.

In Fig.9, we represent the mean confusion matrix (expressed in percent of the sounds of the original class) for the 6 experiments of the LODO evaluation method. The last column of the figure represents the total number of sounds used for each instrument class. Clearly visible in the matrix, is the low confusion between sustained and non-sustained sounds. The largest confusions occur inside each instrument family (viola recognized at 37% as a cello, violin at 14% as a viola and 16% as a cello, French-horn at 23% as a tuba, cornet at 47% as a trumpet, English-horn at 49% as a oboe, oboe at 20% as a clarinet). Note that the classes with the smallest recognition rate (cornet at 30% and English-horn at 12%) are also the classes for which the training set was the smallest (53 cornet sounds and 41 English-horn sounds). More surprising are the confusions inside the non-sustained sounds (piano recognized as guitar or harp, guitar recognized as cello-pizz). Cross-family confusions as the trombone recognized at 12% as a bassoon, recorder recognized at 10% as a clarinet or clarinet recognized at 23% as a flute can be explained perceptually (we have considered a large pitch range for each instrument, therefore

Table 4: Instrument name abbreviations used in Fig.7

Abbreviation	Instrument name
PIAN	Piano
GUI	Guitar
HARP	Harp
VLNP	Violin pizz
VLAP	Viola pizz
CELLP	Cello pizz
DBLP	Double pizz
VLN	Violin
VLA	Viola
CELL	Cello
DBL	Double
TRPU	Trumpet
CORN	Cornet
TBTB	Trombone
FHOR	French-horn
TUBB	Tuba
FLTU	Flute
PICC	Piccolo
RECO	Recorder
CLA	clarinet
SAXTE	Tenor sax
SAXAL	Alto sax
SAXSO	Soprano sax
ACC	Accordeon
OBOE	Oboe
BSN	Bassoon
EHOR	English-horn

the timbre of a single instrument can drastically change).

ACKNOWLEDGEMENT

Part of this work was conducted in the context of the European I.S.T. project CUIDADO [28] (http://www.cuidado.mu).

Results obtained using CFS algorithm, C4.5. and PART have been done with the open source software Weka (http://www.cs.waikato.ac.nz/ ml/ weka) [6].

Thanks to Thomas Helie, Perfecto Herrera, Arie Livshin and Xavier Rodet for fruitful discussions.

8. REFERENCES

[1] G. Ballet. Studio online, 1998.

[2] G. Box and D. Cox. An analysis of transformations. *Journal of the Royal Statistical Society*, pages 211–252, 1964.

[3] J. Brown. Computer identification of musical instruments using pattern recognition with cepstral coefficients as features. *JASA*, 105(3):1933–1941, 1999.

[4] A. Eronen. Comparison of features for musical instrument recognition. In *WASPAA (IEEE Workshop on Applications of Signal Processing to Audio and Acoustics)*, New York, USA, 2001.

[5] J. Foote. *Decision-Tree Probability Modeling for HMM Speech Recognition.* Phd thesis, Brown University, 1994.

[6] E. Frank, L. Trigg, M. Hall, and R. Kirkby. Weka: Waikato environment for knowledge analysis, 1999-2000.

[7] E. Frank and I. H. Witten. Generating accurate rule sets without global optimization. In *Fifteenth International Symposium on Machine Learning*, pages 144–151, 1998.

[8] L. Fritts. University of iowa musical instrument samples, 1997.

[9] J. M. Grey and J. W. Gordon. Perceptual effects of spectral modifications on musical timbres. *JASA*, 63(5):1493–1500, 1978.

[10] M. Hall. Feature selection for discrete and numeric class machine learning. Technical report, 1999.

[11] P. Herrera, A. Dehamel, and F. Gouyon. Automatic labeling of unpitched percussion sounds. In *AES 114th Convention*, Amsterdam, The Nederlands, 2003.

[12] P. Herrera, G. Peeters, and S. Dubnov. Automatic classification of musical instrument sounds. *Journal of New Musical Research*, 2003.

[13] K. Jensen and K. Arnspang. Binary decision tree classification of musical sounds. In *ICMC*, Bejing, China, 1999.

[14] J. Krimphoff, S. McAdams, and S. Windsberg. Caractrisation du timbre des sons complexes. ii: Analyses acoustiques et quantification psychophysique. *Journal de physique*, 4:625–628, 1994.

[15] A. Livshin, G. Peeters, and X. Rodet. Studies and improvements in automatic classification of musical sound samples. In *submitted to ICMC*, Singapore, 2003.

[16] K. Martin. *Sound source recognition: a theory and computational model.* Phd thesis, MIT, 1999.

[17] K. Martin and Y. Kim. 2pmu9. instrument identification: a pattern-recognition approach. In *136th Meet. Ac. Soc. of America*, 1998.

[18] S. McAdams, S. Windsberg, S. Donnadieu, G. DeSoete, and J. Krimphoff. Perceptual scaling of synthesized musical timbres: common dimensions specificities and latent subject classes. *Psychological research*, 58:177–192, 1995.

[19] Microsoft. Musical instruments cd-rom.

[20] J. R. Miller and C. E. C. Perceptual space for musical structures. *JASA*, 58:711–720, 1975.

[21] L. Molina, L. Belanche, and A. Nebot. Feature selection algorithms: A survey and experimental evaluation. In *International Conference on Data Mining*, Maebashi City, Japan, 2002.

[22] MPEG-7. Information technology - multimedia content description interface - part 4: Audio, 2002.

[23] F. Opolko and J. Wapnick. Mcgill university master samples cd-rom for samplecell volume 1, 1991.

[24] G. Peeters, S. McAdams, and P. Herrera. Instrument sound description in the context of mpeg-7. In *ICMC*, Berlin, Germany, 2000.

[25] G. Peeters and X. Rodet. Automatically selecting signal descriptors for sound classification. In *ICMC*, Goteborg, Sweden, 2002.

[26] J. R. Quinlan. *C4.5.: Programs for machine learning.* Morgan Kaufmann, San Mateo, CA, 1993.

[27] E. Scheirer and M. Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. In *ICASSP*, Munich, Germany, 1997.

[28] H. Vinet, P. Herrera, and F. Pachet. The cuidado project. In *ISMIR*, Paris, France, 2002.

[29] E. Wold, T. Blum, D. Keislar, and J. Wheaton. Classification, search and retrieval of audio. In B. Furth, editor, *CRC Handbook of Multimedia Computing*, pages 207–226. CRC Press, Boca Raton, FLA, 1999.

[30] H. Zhang. Heuristic approach for generic audio data segmentation and annotation. In *ACM Multimedia*, Orlando, Florida, 1999.

sust/non-sust	non-sust	pluckstring	pizzstring	sust	bowedstring	brass	reedair	reedsingledouble
temporal increase *temporal decrease* *temporal log-attack*	*temporal decrease* *temporal centroid*			*temporal decrease*				temporal decrease
spectral centroid spectral spread	spectral centroid spectral spread spectral skewness	spectal spread +std spectral slope spectral variation + std	spectral spread sharpness spectral kurtosis spectral slope spectral variation	spectral spread spectral skewness spectral kurtosis + std spectral variation spectral decrease std	spectral centroid spectral spread sharpness spectrall skewness std spectral kurtosis	spectral centroid spectral skewness spectrall kurtosis std	spectral skewness spectral kurtosis + std spectral slope spectral variation std	spectral centroid spectral spread spectral skewness
harmonic deviation		tristimulus + std	tristimulus	*harmonic deviation*	tristimulus	noisiness	harmonic deviation tristimuls std	tristimulus harmonic deviation
mfcc2,6 std	various mfcc	various mfcc	various mfcc		mfcc3,4,6	xcorr 3, 6, 8	xcorr3	xcorr3

Fig. 8: Main selected features by the IRMFSP algorithm at each node of the H-GC tree

		classified as																						
		piano	guitar	harp	viola-pizz	bass-pizz	cello-pizz	violin-pizz	viola	bass	cello	violin	french-horn	cornet	trombone	trumpet	tuba	flute	piccolo	recorder	bassoon	clarinet	english-horn	oboe	number of sounds
real class	piano	36	29	24	1	1	2	3			1		1				2				1	1			146
	guitar	3	48	22		3	20	1													4				159
	harp	4	12	68	6		2			2							2				5				130
	viola-pizz	4		2	85		4	6																	**54**
	bass-pizz	2	1	3		76	18															1			186
	cello-pizz	2	8	5	1	12	71	1																	170
	violin-pizz	1		2	9			88																	97
	viola								44		37	14										2			225
	bass									93	5									1		1			280
	cello						1		5	4	68	3		2		1		1			1	7	1	4	356
	violin	1							14	6	16	55	1					2				2	1	1	264
	french-horn										1		50	1	15		23	5			2		1		242
	cornet													30	15	47		2			2			4	**53**
	trombone												13	3	49	10	7	3			12				202
	trumpet												1	13	7	61		4	1			4		9	157
	tuba	2					1			1			15				79	1							140
	flute							2		2	2	1		2	1			77	4	2		5		3	323
	piccolo				4			4					4	1	1			10	71	4		1		1	83
	recorder												5			3		10	5	59	3	10	3	3	39
	bassoon				2			3		1	1		2		2			1			81		3	1	203
	clarinet									3	1					2		23	5			46	1	14	212
	english-horn										5	10						2			12	10	12	49	**41**
	oboe											1		4		2		4	8		1	20	4	58	184

Fig. 9: Overall confusion matrix (expressed in percent of the sounds of the original class) for the LODO evaluation method. Thin lines separate the instrument families while thick lines separate the sustained/non-sustained sounds.