MUSICAL SIGNAL TYPE DISCRIMINATION BASED ON LARGE OPEN FEATURE SETS

Björn Schuller, Frank Wallhoff, Dejan Arsić, and Gerhard Rigoll

Institute for Human-Machine Communication
Technische Universität München, Germany
{Schuller; Wallhoff, Arsic, Rigoll}@tum.de

ABSTRACT

Automatic discrimination of musical signal types as speech, singing, music, genres or drumbeats within audio streams is of great importance e.g. for radio broadcast stream segmentation. Yet, feature sets are largely discussed. We therefore suggest a large open feature set approach starting with systematical generation of 7k hi-level features based on MPEG-7 Low-Level-Descriptors and further feature contours. A subsequent fast Gain Ratio reduction followed by wrapper-based Floating Search leads to a strong basis of relevant features. Next, features are added by alteration and combination within genetic search. For classification we use Support-Vector-Machines proven reliable for this task. Test-runs are carried out on two task-specific databases and the public Columbia SMD database and show significant improvements for each step of the suggested novel concept.

1. INTRODUCTION

A great interest exists in the automatic discrimination of audio signal types as speech, music, speech overlaid music, acapella singing, musical genres or drum-beats. E.g. automatic speech recognition applied to soundtracks [1] demands segmentation between music and speech parts prior to speech recognition. In radio streams parts of the D.J. can likewise be excluded or exclusively retrieved [2]. Discrimination of monophonic singing and speech can be applied in music retrieval interfaces controlled by both interaction forms prior to speech or singing analysis [3]. Likewise a user does not need to indicate e.g. manually whether singing or speaking at a specific time. Finally, in Query-by-Singing applications matching to the polyphonic original audio it may be useful to retrieve singing locations or such containing merely drumbeats, to find reference parts of the key-melody. Finally, musical genre type discrimination has many commercially interesting applications as automatic equalizer adjustment or sorting of musical databases.

So far several works deal with the discrimination of polyphonic music and speech [1, 2, 4], while rather few work on the harder challenge of discrimination between speech and monophonic singing [5] or singing location [6], in our case even of the same person [7]. In these works rather low numbers of features have been considered, and selected by single feature relevance calculation instead of finding an optimal set which is also ideally suited for the target classifier.

Herein we strive to improve on this task by introduction of large open feature sets of 7k+ features. A systematic generation of features based on Low-Level-Descriptors found in the MPEG-7 standard and further ones forms the feature basis. Afterwards a fast pre-selection of relevant ones by filter search takes place. Next, we optimize a set by floating wrapper search. Finally, we allow for feature alterations and cross-feature analysis by use of Genetic Algorithm based feature generation.

As prove of concept extensive test-runs on three partially public databases shall demonstrate effectiveness of the suggested approach.

The paper is structured as follows: In section 2 we describe applied databases. Sections 3 and 4 deal with extraction, pre-selection and classification of acoustic features. In section 5 we discuss automatic feature generation by evolutionary programming. The final two sections discuss results obtained and show future directions.

2. DATABASE DESCRIPTION

Firstly, we use the public Columbia University Speech Music Discrimination (SMD) database introduced and used in [1,8]. This database contains among other samples segments from a radio broadcast stream. We use the total of 101 samples of music, 80 of speech, and 60 of music overlaid with speech contained in this database herein.

Secondly, we extend our previously introduced SHANGRILA corpus of speech and monophonic singing samples [9]. It comprises of 1,000 samples of speech and 1,114 samples of singing of 58 persons in total. These audio samples have been recorded in 16bit, 11 kHz by use of an AKG MK 1000S-II condenser microphone. They resemble interaction turns with a music retrieval interface as introduced in [3]. Polyphonic music clips are taken from 200 songs of the MTV-Europe-Top-10 of the years 1981-2000. The clips were cut out at five fixed relative positions of each song resulting in 1,000 clips in total. The genres covered resemble typical mainstream pop-music radio station sound. Additionally, we added 1,000 drum beat clips that consist of various styles as disco, jazz, rock, and techno music. The whole corpus is abbreviated SAB in the ongoing. By this second database we can show results on a higher total of samples and for further audio signal types.

Thirdly, we introduce a database for musical genre discrimination named GeDi to evaluate effectiveness of the proposed method on this task. 6 genres are covered by 602 tracks: Classical Music (collection *"100 Meisterwerke der klassischen Musik"*, 6 CDs, 100 tracks), Electronic Dance-Music (collections *"Future Trance vol. 32", vol. 33*, and *vol. 34*, 6 CDs, 126 tracks), Jazz (collection "Blue Note Jazz History", 5 CDs, 106 tracks) Rock Music (collections *"Best of Rock"*, 1 CD, *"Driving Rock"*, 2

CDs, *"Fetenhits - Rock Classics"*, 2 CDs, *"Rock Super Stars Vol. 3"*, 1 CD, 99 tracks in total), Live Music (99 Songs randomly selected from diverse interprets, albums, songs, no doubles), and audio-documents (71 randomly selected pieces from comedians and ear-books, no doubles in view of recording) . These genres have been selected having a car-stereo or portable MP3-device in mind that shall be enabled to automatically adjust equalizer settings in accordance with typical equalizer presets or sort playlists.

3. LARGE FEATURE SET CONSTRUCTION

We use systematic generation of functionals *f* out of time-series *F* by means of descriptive statistics:

$$f : F \rightarrow \mathbb{R} \quad (1)$$

Firstly, selected base-contours, respectively Low-Level-Descriptors (LLD), are calculated well known to carry information about the musical signal type. The original sampling frequency and quantization of the databases is kept, and each 10 ms a 20 ms frame is extracted by weighting with a Hamming window-function. Aiming at broad coverage, estimated feature contours contain log frame energy, pitch based on autocorrelation (ACF) in the time-domain and Dynamic Programming (DP) to minimize deviations on a global level, pitch epochs, harmonics-to-noise ratio based on ACF, formant bandwidth, position and amplitude of the first 5 formants based on LPC, polynomial roots and DP. Further more jitter and shimmer is calculated. Thereby jitter is a measure of pitch- and shimmer one of amplitude-perturbation on a cycle to cycle basis. For spectral analysis 16 MFCCs, and spectral flux, spectral centroid, as well as spectral roll-off based on linear DFT-spectral coefficients and polynomial dB-correction in accordance to human perception, are extracted. Dominant harmonics in the spectrum are tracked in 47 chromatic semitone intervals within human voice range by summing over three successive partials. Finally, 19 Voc19 coefficients are obtained by JSRU-style 19-channel filter-bank analysis using two second-order section Butterworth band-pass filters. Energy smoothing is done at 50Hz.

The contours are subsequently smoothed by symmetrical moving average low-pass filtering with a window size of three. Likewise we are less prone to noise throughout the calculation, as most feature contours as pitch or formants are prone to errors, already. Successively, speed (∂) and acceleration (∂^2) are derived as further LLDs for each basic contour in order to model temporal behavior.

Afterwards a total of 21 clip-wise derived hi-level functionals by means of descriptive statistics per contour is computed. One exception is the genre discrimination task where hi-level features are computed for seconds 0-3, for seconds 25-28, and as global statistics over these two parts per song. Afterwards a super-vector is constructed per song within this task. This is done having an audio buffer of 30 sec in mind and leaving 2 sec prior to song change for processing in the intended equalizer adjustment scenario. The derived attributes are linear momentums of the first four orders, namely mean, centroid, standard deviation, Skewness and Kurtosis, as well as quartiles, quartile ranges, extremes, extreme positions, range, zero-crossing-rates, 95%-roll-off-points, 25%-down-level-time, and 75% up-level-time. Likewise roughly 7k acoustic features are obtained in total. The aim here is too build a broad feature basis for the subsequent feature selection process, throughout which is learned which attributes to prefer in which scenario. Thereby almost redundant features are justified at this stage.

4. PRE-SELECTION AND CLASSIFICATION

Besides lower extraction time-effort, reduction of features also often leads to higher classification performance, as the classifier is confronted with less complexity, if only redundant information is spared. In former works [9] we demonstrated the high effectiveness of wrapper-based search which aims at optimization of a set as a whole. However, due to the unusually high dimensionality in this domain of 7k entries in the original feature vector we apply fast Information Gain Ratio based feature selection (IGR-FS) herein, firstly. In this filter-reduction single highly relevant attributes are found by their entropy [11]. Likewise, ranking of attributes is independent of the classifier. However, we use a closed feed-back loop in order to find the optimal number of the ranked features in accordance with the target classifier.

After such pre-reduction to the optimal feature set size by IGR-FS we apply the more powerful Sequential-Forward-Floating-Search (SFFS) to further reduce feature set size and raise accuracy by less complexity for the classifier. SFFS is a Hill-Climbing search that starts with an empty feature set and measures feature relevance by classification accuracy. Iteratively new features are added to the set. Backward steps in a floating manner help to cope with nesting effects.

Dealing with classification, the optimal learning method is broadly discussed, similar to the optimal features. In [9] we made an extensive comparison on the SHANGRILA database including besides Support Vector Machines (SVM), Naïve Bayes, k-Nearest Neighbors, Decision Trees, and Neural Nets. Further more we investigated construction of more powerful classifiers by means of meta-classification as MultiBoosting or Stacking. However, in our experiments SVM prevailed as base classifiers. We therefore concentrate on these herein.

SVM - kernel machines - are well known in the machine learning community and highly popular at the time due to their remarkable performance and generalization capabilities. Generally speaking, SVM base on a linear distance-function classification of a two-class problem. However, multi-class strategies as one-vs.-one, layer-wise decision or one-vs.-all exist. Discriminative training is achieved by optimal placement of a separation hyperplane under the precondition of linear separability which is approached by a transformation of the original feature space via a kernel function that has to be found empirically.

In this evaluation we use a couple-wise one-vs.-one decision for multi-class discrimination and a polynomial kernel found optimal throughout test cycles. For more details on classifiers refer to [11].

5. EVOLUTIONARY GENERATION

Besides reduction of the feature space, also its expansion can lead to improved accuracy. Consider hereon the Kernel-trick in SVM classification. However, while an optimal Kernel has to be selected empirically, we aim at a self-learning approach to feature space transformation based on random injection. Especially the combination of both by a suited search algorithm and the target classifier, allows for self-learning optimization of the ideal

representation within feature space. In order to expand the feature space we generate novel features based on the so far pre-selected ones: Firstly, alteration of attributes by mathematical operations can be performed to lead to better representations of these. Consider hereon the standard use of logarithmic HNR representation. So far we only considered features based on single contours. By association of these we can secondly obtain a further number of new information as inter-band dependencies. As a deterministic and systematic generation comes to its limits applying exhaustive search, we decided for Genetic Algorithm (GA) based search through the possible feature space. The parallel selection of most relevant information and reduction is fulfilled within one pass by this GA based search.

GA, a well-known bio-analog method, base on Darwin's *survival-of-the-fittest* principle of mutation and selection [12]. We also include crossing of parental DNA information - in our case feature crossing. GA are computationally expensive, but they can be parallelized to a high degree.

The precondition is to have a start-set of effectually different individuals that represent possible solutions to the problem. In our case these are partitions of the acoustic feature sets reduced to a reasonable size by now. The partitions are denoted in binary coding, and are called *chromosomes* in terms of GA literature. Each chromosome consists of *genes* that correspond to single features within the partition. A feature's gene consists of one bit for its activity status. The partitioning is done randomly throughout initialization and we obtain $N=\dim(\underline{x})/n$ individuals if $\underline{x}$ denotes the feature vector, and n the partition size.

By an initialization probability, set to 0.5 in our case, it is randomly decided which original features are chosen for one step of genetic generation. We decided to have a *population* size of 20 individuals at a time. Next a *fitness* function is needed in order to decide which individuals survive. Thereby the aimed at classifier forms a reasonable basis in view of wrapper based set optimization. A cyclic run over multiple *generations* is afterwards executed until an optimal set is found, which resembles a local maximum of a problem:

Firstly, a *Selection* takes place, based on the fitness of an *individual*. We use common *Roulette Wheel* selection within this step. Thereby the 360° of a roulette wheel are shared proportional to the fitness of an individual. Afterwards the "wheel" is turned several times, resembling N times selecting out of N individuals. Selected individuals are assembled in a *Mating Pool*. Likewise, fitter individuals are selected more probably. We also ensure mandatory selection of the best one, known as *Elitist Selection*.

The oncoming *Crossing* of pairs is fulfilled by picking $N/2$ times individuals with the probability $1/N$. After selection, individuals are put aside. Opposing traditional GA, we use a variable chromosome length from hereon, as we aim at generation of features. First we have to pick to *parents* in order to cross their chromosomes and thereby obtain new *children*. Thereby the distance between parents and children should reasonably be smaller than the one between parents themselves. We therefore choose simple *Single-Point-Crossing* which splits each parent chromosome close to its center and pastes the two halves cross-wise to obtain two children. The fitness thereby also limits the total number of children an individual may produce.

Afterwards, *Mutation* takes place: the state of a gene, respectively of a feature within a partition, is randomly changed by a probability of 0.5. Likewise features can be excluded from a set. To generate new feature we insert a random selection of an alteration method out of *reciprocal value*, *addition*, *subtraction*, *multiplication* and *division* [12]. Depending on the mathematical operation the appropriate number of features within an individual is selected for alteration, and the operation is performed. Thereby new features can be constructed by combination of original ones. The obtained new individuals are than appended within the chromosome.

Now the Evaluation of the population is fulfilled, which resembles the fitness-test – in our case classification with the feature sub-sets. We use SVM on cross-validation set, as we want to optimize the feature space for SVM classification. At this point, one iteration is finished, and the algorithm starts over with Selection. We decided for a maximum of 50 generations, and 40 of them without improvement. To conclude the feature extraction, selection and classification process so far, figure 1 provides a general overview.

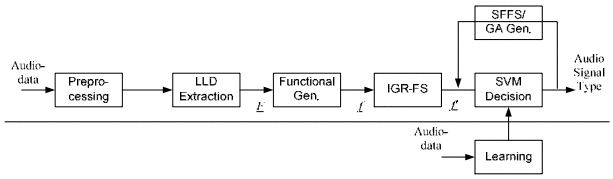

Figure 1: *Overview audio signal type discrimination.*

6. RESULTS

Within this section we present results of test runs on the described databases within 10-fold stratified cross-validation (SCV).

The first table shows results for the stepwise optimization of the feature space by using all features, subsequent IGR-FS, SVM-SFFS, and finally addition of genetic generation by GA.

Table 1: Error signal type discrimination using SVM and stepwise feature-space optimization, 10-fold SCV.

Error [%]	**SMD**	**SAB**	**GeDi**
All features	8.4	9.2	12.8
-IGR-FS	5.7	5.0	11.6
-SVM-SFFS	3.7	3.1	7.8
+GA Generation	3.3	2.1	7.8

As can be seen, all steps lead to a significant improvement on error rates based on a paired Student-T-Test and a significance level of $\alpha=0.05$ besides genetic generation on GeDi database. Likewise, besides mere reduction of the feature space, also generation of novel features based on the original ones may help to improve on error rates. We therefore call the feature sets *open*. For the public database SMD the features were reduced starting from 7k to 197 by IGR-FS, afterwards to 76 by SVM-SFFS. By genetic generation 7 new features could be added basing on these. On GeDi features were reduced to 500 by IGR-FS and to 92 by SVM-SFFS. In a similar relation features were reduced and generated for SAB.

Within the next table 2 class-wise mean error rates are presented for the SMD database. As samples are not evenly distributed among classes, we also show each class's F_1-Measure. Music is recognized the worst, while music overlaid with speech *(Mu+Sp)* is recognized the best.

Table 2: Class-wise error and F_1-Measure, database SMD, optimal features, SVM, 10-fold SCV.

[%]	Speech	Music	Mu+Sp
Error	2.5	4.0	1.7
F_1-Measure	98.1	97.0	95.2

Table 3 depicts the class-wise mean error-rates for the database SAB. As all samples are evenly distributed among classes, F-Measures are spared.

Table 3: Class-wise error, database Shangrila+Beat, optimal features, SVM, 10-fold SCV.

SAB	Speech	Music	Singing	Beat
Error [%]	0.3	0.1	1.4	6.8

Drumbeats (Beat) are detected least best. This is due to confusions with polyphonic music. On the other hand side, music is practically not confused with beat.

For the GeDi database 14.5% error are observed omitting global features, 20.8% error using exclusively seconds 0-3, 28.1% error using exclusively seconds 25-28, and 15.6% using exclusively global statistics of the combination of these two clips. Likewise the first three seconds seem more important, but the best result is obtained by the propagated strategy, namely 7.8% error. Table 4 depicts class-wise accuracies.

Table 4: Class-wise error and F_1-Measure database GeDi, optimal features, SVM, 10-fold SCV. A.doc= Audio Document, Class.=Classical Music, Dance= Electronic Dance-Music

[%]	A.doc	Class.	Dance	Jazz	Live	Rock
Err.	2.8	0.0	11.1	7.5	2.0	20.8
F_1-M.	97.9	99.5	88.9	91.6	95.6	82.1

If the classes with highest confusion, i.e. rock vs. electronic dance music, are left out, the error rate sinks as low as 0.8% using the optimal configuration. The error for mere separation between rock music and live rock music resembles 4.0%.

7. CONCLUSIONS

Within this contribution we showed a novel approach to audio signal type discrimination by large open feature sets. Based on Low-Level-Descriptors 7k hi-level-features are derived by means of descriptive statistics. In order to cope with this high complexity and find task specific relevant ones, feature selection methods were applied. Firstly, a fast filter-based pre-selection finds generally suited features, next a more compact optimized set is found by wrapper selection. The open character is realized by evolutionary feature generation based on feature alteration and cross-feature attribute construction. Significant improvements within every step could be demonstrated on three test-sets. Overall achieved error rates are outstandingly low: besides the discrimination of speech, music, monophonic singing, and music overlaid with speech also drum beats and six musical genres could be recognized.

In future works we aim at integration of these concepts in existent Query-by-Singing approaches. Further more the general principle may be applied in related audio signal type discrimination tasks as musical mood recognition. A further refinement of hi-level feature generation may thereby lead to improved performance as well as diverse timing-levels.

8. REFERENCES

[1] E. Scheirer, M. Slaney: "Construction and evaluation of a robust multifeature speech/music discriminator," *Proceedings of the ICASSP 97*, pp. 1331–1334, 1997.

[2] J. Saunders: "Real time discrimination of broadcast speech/music," *Proceedings of the ICASSP 96*, pp. 993-996, 1996.

[3] B. Schuller, M. Zobl, G. Rigoll, M. Lang: "A Hybrid Music Retrieval System using Belief Networks to Integrate Queries and Contextual Knowledge," *Proceedings of the ICME 2003*, Baltimore, MD, USA, Vol. I, pp. 57-60, 2003.

[4] W. Chou, L. Gu: "Robust Singing Detection in Speech/Music Discriminator Design," *Proceedings of the ICASSP 2001*, 2001.

[5] D. Gerhard: "Pitch-based acoustic feature analysis for the discrimination of speech and monophonic singing," *Journal of the Canadian Acoustical Association* 30:3, pp. 152-153, 2002.

[6] A. L. Berenzweig, D. P. W. Ellis: "Locating Singing Voice Segments Within Music Signals," *Proceedings of the WASPAA 2001*, Mohonk, NY, USA, 2001.

[7] B. Schuller, G. Rigoll, M. Lang: "Discrimination of Speech and Monophonic Singing in Continuous Audio Streams Applying Multi-Layer Support Vector Machines," *Proceedings of the ICME 2004*, Taipei, Taiwan, 2004.

[8] G. Williams, D. Ellis: "Speech/music discrimination based on posterior probability features," *Proceedings of the Eurospeech 99*, Budapest, Hungary, 1999.

[9] B. Schuller, B. J. Brüning Schmitt, D. Arsic, S. Reiter, M. Lang, G. Rigoll: "Feature Selection and Stacking for Robust Discrimination of Speech, Monophonic Singing, and Polyphonic Music," *Proceedings of the ICME 2005*, IEEE, Amsterdam, Netherlands, 2005.

[10] B. Logan, "Mel Frequency Cepstral Coefficients for Music Modeling," *Proceedings of the International Symposium on Music Information Retrieval ISMIR 2000*, 2000.

[11] I. H. Witten, E. Frank, Data Mining, *Practical machine learning tools with Java implementations*, Morgan Kaufmann, San Francisco, pp. 133, 2000.

[12] D. E. Goldberg: *Genetic Algorithms in Search, Optimization & Machine Learning,* Addison-Wesley Publishing Company, Inc., 1989.

[13] I. Mierswa: "Automatic Feature Extraction from Large Time Series," *Proceedings of the 28. Annual Conference of the GfKl 2004*, Springer, pp. 600-607, 2004.