Perceptually Based Phoneme Recognition in Popular Music

Gero Szepannek[1], Matthias Gruhne[2], Bernd Bischl[1], Sebastian Krey[1], Tamas Harczos[2], Frank Klefenz[2], Christian Dittmar[2], and Claus Weihs[1]

[1] Faculty of Statistics, Dortmund University of Technology
szepannek@statistik.tu-dortmund.de
[2] Fraunhofer Institute of Digital Media Technology (IDMT)

Summary. Solving the task of phoneme recognition in music sound files may help for several practical applications: it enables lyrics transcription and as a consequence could provide further relevant information for the task of an automatic song classification. Beyond it can be used for lyrics alignment e.g. in karaoke applications. The effect of both different feature signal representations as well as the choice of the appropriate classifier are investigated. Besides, a unified R framework for classifier optimization is be presented.

Key words: Music, Feature Extraction, Auditory Models, Classifier Optimization

1 Introduction

This work is an extension of previous studies at the Fraunhofer IDMT on automatic phoneme classification in polyphonic music [7]. An accurate phoneme recognition in music may yield a basis for several applications like automatic lyrics extraction (and further automatic classification of songs) as well as for the automatic alignment of previously known lyrics to music for karaoke applications. The specific goal of this work consists of the examination of different feature sets extracted from audio data combined with an appropriate choice and parameter tuning of classifiers. Concerning the feature sets, perceptive phenomena have been more and more introduced in audio processing during the last years. It is of interest up to what extent it is beneficial to model human sound processing. A detailed neurophysiological simulation model of the human auditory periphery (serving as basis for feature extraction) is compared to a simpler and computationally less expensive phenomenological one. Auditory model-based features are opposed to well-known standard acoustical feature vector representations. A unified framework for the statistical programming language R is presented that easily allows to tune, optimize and compare the influence of different classifiers for specific data situations and the given task.

A description of the task in Section 2 is followed by a brief introduction to auditory modelling in Section 3. Feature extraction from audio data (based on the original waveform as well as on the auditory simulation model output) is described in Section 4. The framework for classifier optimization is presented in Section 5. Finally, the results of the study, a discussion and a summary are given in Sections 6 and 7.

2 Description of the Task

The data under investigation consists of 45 files of popular music (30 male and 15 female singers, 44.1 kHz, e.g. *I'm a believer* (Monkees), *Sweet dreams* (Eurithmics), *Zepher* (Red Hot Chili Peppers), *Billie Jean* (Michael Jackson), *Killing me softly* (Roberta Flack), *Song* #2 (Blur), ...). The music files are split into training and test files ($2/3 : 1/3$). All songs are phonetically manually labelled at the Fraunhofer IDMT according to the TIMIT phonetic transcription. Only one single feature vector (from a 1024 samples window, i.e. 23.6 ms) is computed per phoneme to avoid highly correlated observations. In automatic speech recognition monophones are typically modelled as three state hidden Markov models (HMMs) where the second state corresponds to its stationary part (see e.g. [6], p. 365). We assumed to maximise the chance to hit this "inner" steady state of the phonemes when considering the window at half of the phonemes total duration. 15 different vowels as well as consonants were taken into the analysis as far as there were at least 50 observations of each phoneme in total. The resulting classes are: */a/, /ae/, /e/, /ee/, /i/, /j/, /l/, /m/, /n/, /o/, /oa/, /oe/, /ou/, /r/ and /w/*. Finally, there were 1549 phonemes in the training and 672 phonemes in the test set. For the detailed auditory model the `*.wav` files have to be amplitude normalized before processing to be able to set the absolute sound pressure level (SPL) (see also Sec. 3).
We applied sinusoidal preprocessing as it turned out to be beneficial [7]. Basically, the audio signal is considered to be a sum of voice and background. The voiced part is further modelled as a sum of sinusoids of the estimated fundamental and its harmonics. The amplitudes of all other fourier frequencies are set to 0 in the spectral domain and the result is back-transformed to the time domain (for further details, see [7]).

3 Auditory Modelling

t

Fig. 1. Tuning curve for a 1000 Hz sine sound of different sound pressure level (left) and output of the auditory model for a vowel /a/ (right).

Several well-known psycho acoustical phenomena can be traced back to sound processing in the auditory system, e.g. nonlinear frequency resolution and amplitude saturation or masking effects. Basically, the sound wave is nonlinearly bandpass-filtered at the inner ear along the *basilar membrane* (BM) and transduced into electric impulses (action potentials, APs) at the *auditory nerve fibres* (ANFs) of different *center frequency* (CF) by *inner hair cells*. A simple computational auditory model (referred to as "Seneff-model", [11]) phenomenologically imitates human auditory sound processing within a chain of five successive steps, consisting of: critical band filtering *(BM excitation)*, halfway rectification saturating non linearity *(inner hair cell current)*, short term adaption circuit *(synaptic neurotransmitter release)*, low pass filtering *(nerve fibre: synchrony reduction)* and rapid automatic gain control *(nerve fibre: refractory effect)*. The output of the model can be interpreted as *time varying neural firing rates* at 40 different ANFs (of 0.5 bark CF difference).
Besides this, also a very detailed and computationally more intensive model of the human auditory periphery is implemented where all steps reproduce neurophysiological measurements. It simulates exact firing times of 251 different ANFs with CF differences of 0.1 bark [12].
Figure 1 (left) shows the average auditory nerve firing activity of the detailed model during 200 ms along the BM (abscissa) for a 1 kHz sine of different sound pressure level (SPL, ordinate). A level of 0 dB SPL denotes the threshold of hearing [6]. Figure 1 (right, bottom) shows the response of the auditory simulation model to some vowel /a/. The ordinate represents the unrolled inner ear (BM) while the abscissa denotes the time. The output of the simulation model is binary and of the form

$$X_i(t) = \begin{cases} 1, \text{ AP of ANF } i \text{ at time } t, \\ 0, \text{ else.} \end{cases}$$

It can be seen that different positions along the BM are differentially excited (according to the signal frequencies). The ANFs further respond periodically with the signal period. This phenomenon is commonly referred to as *phase locking* [14]. For the studies in this paper, 50 repetitive simulations of the ANFs of different type are pursued. The signals were presented to the auditory model at a (typical) level of 62.5 dB SPL [14].

4 Feature Extraction

A key idea of timbral feature extraction is the *source-filter model* (of speech production). Speech signal waves are excited at the glottis (either noisy or periodic) and get their characteristic timbre being filtered by the specific shape of the vowel tract. Thus, the filter coefficients of (fixed) order p meaningfully represent the sound characteristics. These *linear predictive (filter) coefficients*

(LPCs) are derived by Levinson-Durbin recursion to minimize the predictive error (see e.g. [6]). According to former studies [13] a choice of $p = 16$ is used here.

Based on the principles of neural information coding mentioned above two different non-standard feature sets are extracted from the simulated auditory neural response (see Section 3). *Place / mean rate features* (MR) count the neural activity at different ANFs independently of its temporal fine structure, i.e.

$$X_i^{MR} = \sum_{t \in \text{ window}} X_i(t)/\text{window size}.$$

According to [2] groups of 8 neighbouring ANFs of the detailed auditory model in the CF range of $[200, 6400]$ Hz are averaged to build a 24 dimensional feature vector.

On the other hand, *average localized synchrony detection* features (ALSD, [1]) temporally encode neural auditory information:

$$X_k^{ALSD} = \frac{1}{3} \sum_{l=k-1}^{k+1} A_s \tan^{-1}\left[\frac{1}{A_s}\left(\frac{\langle|X_l^{PSTH}(t) + X_l^{PSTH}(t-n_k)|\rangle - \delta}{\langle|X_l^{PSTH}(t) - \beta^{n_k} X_l^{PSTH}(t-n_k)|\rangle}\right)\right] \tag{1}$$

with $X_l^{PSTH}(t)$ being the time-varying firing rate of ANF l (estimated by the post stimulus time histogram of the neural activity in time bins of $\frac{1}{14700}$ s averaged over all simulations and 8 neighbour ANFs as for X^{MR} for the detailed model). The $\langle . \rangle$ operator denotes temporal averaging, n_i is the period (in time bins) of the CF of ANF i. Basically, the denominator checks, whether on average the neural activity is the same as it has been one (CF-)period before. The constant $\beta = 0.99$ avoids obtaining zeros in the denominator. $\delta = 60\,\text{spikes}\,dt\,s^{-1}$ corrects for spontaneous neural activity and $A_s = 4$ is a scaling constant. According to eqn. (1) the X^{ALSD} representation consists of a 22 dimensional feature vector.

Mel frequency cepstral coefficients (MFCCs) (see e.g. [6], pp. 280-288) have recently become popular for speech and music analysis. They also rely on the source-filter model of speech production: in the spectral domain the signal is the product of the excitation and the filter amplitudes (of the vowel tract). Building the logarithm changes this into a sum. A subsequent inverse discrete fourier transform can be interpreted as a "spectral analysis of the log-spectrum": strong periods in the spectrogram represent the fundamental and its harmonics and are captured in the higher coefficients (*quefrencies*) as well as noise is. The characteristic shape of the log-spectrum is represented in the lower coefficients. Thus, only the lowest q coefficients are used for further timbre analysis. In this application, a typical value of $q = 13$ is chosen. To imitate human perception frequency grouping according to the mel scale is performed. The log transform can be further compared to human auditory nonlinear amplitude saturation [14].

Perceptual linear prediction coefficients (PLPs) also take into account human

auditory sound processing (see [6], p. 299). Before computing LPCs (see above) the sound signal is transformed into the frequency domain where amplitudes are compressed (typically by building cubic roots) and frequencies are grouped according to the perceptive mel scale. After some inverse back-transform into the time domain, LPCs are calculated. Also, an order of $p = 16$ is chosen for this work [13]. For standard features like MFCCs, LPCs and PLPs an `R` implementation of the Matlab `rastamat` toolbox [5] is used.

5 Classifier Tuning

The aim of this work is to investigate the combination of both feature extraction and the choice of an adequate classifier. There exist numerous different classification algorithms (for an overview see e.g. [8]), many requiring the choice of additional free parameters. The list below shows the classifiers that were implemented for this study (in brackets the parameters that were varied): *SVMs with polynomial kernels* ($K(x,y) = (1 + \langle x, y\rangle)^d$, **PSVM**, $d \in \{1,2,3,4\}$, cost of constraints violation $c \in 2^{\{-4,-3,\ldots,3,4\}}$), *SVMs with RBF kernels* ($K(x,y) = e^{-||x-y||^2/\gamma}$, **RSVM**, $\gamma \in 2^{\{-4,-3,\ldots,3,4\}}$), cost $c \in 2^{\{-4,-3,\ldots,3,4\}}$), *linear discriminant analysis* (**LDA**, -), *quadratic discriminant analysis* (**QDA**, -), *mixture discriminant analysis* (**MDA**, equal number of subclasses $\in \{2,\ldots,5\}$), *naive Bayes* (**NB**, -), *classification trees* (**RPART**, factor of required improvement for a split to be kept in the tree model $\in \{0.005, 0.01, 0.03\}$), *random forests* (**RF**, -) and k *nearest neighbours* (k**NN**, $k \in \{1,2,3,4,6,8,10\}$). All classifiers are evaluated in `R` using the packages `kernlab`, `MASS`, `e1071`, `mda`, `rpart`, `randomForest` and `kknn`. The free parameters are optimized on grids using an internal 5-fold cross validation (cv) on the training data. A typical problem using the programming language `R` for classification purposes is the heterogenity of the different implemented algorithms. A framework has been developed in order to easily enable optimizing and benchmarking different classifiers using the `R` package {`mlr`} [3]. Its features are: an object oriented `S4` interface to `R` classification methods, easy extension to new methods, it provides a unified call of different methods, bootstrapping, cross-validation, train/test splits, parameter tuning and benchmarking of different classification algorithms are possible (e.g. by 'double cv' with tuning on an inner cv).

6 Results and Discussion

At first glance, the choice of the feature set appears to play the dominant role on the accuracies. Figure 2 (left) shows the performances as a function of the choice of the feature set. As simple LPCs show the worst results on average it turns out to be worth including perceptive phenomena into feature extraction

design. No strong advantages are observed using the detailed neurophysiologically parameterized auditory model instead of the simpler phenomenological one. Concerning auditory model based feature extraction ALSD outperforms MR feature extraction. Nevertheless standard features like MFCCs and PLPs show the best results. Even their averaged accuracies over all classifiers are better than the best results for optimized classifiers for any of the auditory model based feature sets. It should be noted that these feature sets both include simplified perceptual models as well as a speech production based motivation. The auditory model based features on the other hand are restricted to modelling perception. An additional cepstral transformation of the auditory model based features did not improve the results. There is no equivalent to masking effects included in MFCC or PLP feature extraction. Apparently, this effect is not of relevance for this application. Figure 3 shows the performance of the classification algorithms compared to the average accuracy on each data set separately. The best classifier is not the same for all feature sets emphasizing importance of a problem-specific classifier tuning. Some methods (NB, RPART and kNN) are a bad choice for any of the investigated feature sets. Some other classifiers (SVMs and LDA) are often among the best methods. MDA shows good results for the non-auditorily extracted features whereas random forests only perform well for the auditory model based features.
In order to obtain general hints on the choice of the classifier, a consensus ranking is derived (see e.g. [9]). Significant differences in test accuracy of any two classifiers for each data set are investigated using McNemar's test [4]. The results are summarized using a Bradley-Terry model for paired comparisons. Table 1 shows the resulting π_i that can be interpreted as "prior probabilities" for each specific classifier to be the significantly best choice. The consensus ranking of the classifiers strongly suggests the use of optimized SVMs with polynomial kernel (cf also Figure 3). Nevertheless, the best overall results (53.42%) are obtained using RBF-kernel SVMs and MFCC features once again emphasizing the importance of problem specific classifier choice. Random forests as well as LDA and MDA also appear to be a good choice in general. Nevertheless, generalization of these results should be handled with care. The tuned parameters of the optimal model are $\sigma = 0.0625$ of the RBF kernels and complexity parameter $c = 2$. But the results strongly

Fig. 2. Maximal/average (light/dark grey) accuracy over different classifiers (left). Feature sets clustered according to distances given by the fraction of differently predicted (test-)objects (right)

	PSVM	RSVM	RF	MDA	LDA	QDA	RPART	kNN	NB
π_i	**0.424**	0.155	0.141	0.108	0.099	0.029	0.020	0.015	0.009

Table 1. Consensus ranking of the classifiers over all data sets.

Fig. 3. Classifier performance compared to average accuracy per data feature set.

depend on the parameters, within the investigated parameter grid also accuracies below 20% are observed. Finally, Figure 2 (right) identifies MFCCs and PLPs as well as the simpler auditory (Senneff) features to be more similar to each other than the other features by average linkage clustering. It should be further noted that - in contrast to modelling continuous speech the task of classifying single frames becomes more complicated, especially due to the heterogeneous (nonstationary) polyphonic background noise. The use of HMMs smoothes over successive frames for continuous modelling. The incorporation of posterior probability estimates of optimized classifiers could improve the use of standard Gaussian (MDA like) mixtures for continuous modelling (see e.g. [10]). Further attention could be laid on feature combination as in [14].

7 Summary

A task with many practical applications has been investigated: automatic recognition of phonemes in popular music. Specific interest of the study was the investigation of the influence of different feature representations in combination with the choice and tuning of the appropriate classification method. A new `R` package framework has been presented to solve the latter task. In conclusion both is beneficial: taking into account speech production as well as perception. No improvements have been observed for high degrees of precision in auditory modelling. Nonetheless, the appropriate choice and tuning of the classifier is of importance. The work could be further extended towards feature combination and modelling continuous singing.

References

1. A. Ali, J. van der Spiegel, and P. Mueller. Robust auditory-based speech recognition using average localized synchrony detection. *IEEE Transactions on Speech and Audio Processing*, 10(5):279–292, 2002.
2. J. Allen. How do humans process and recognize speech? *IEEE Transactions on Speech and Signal Processing*, 2(4):567–577, 1994.
3. B. Bischl. The `mlr` package: machine learning in `R`. `http://algorithm-forge.com/bischl/mlr/`, 2010.
4. T. Dietterich. Approximate statistical tests for comparing supervised classification algorithms. *Neural Computation*, 10:1895–1923, 1998.
5. D. Ellis. PLP and RASTA (and MFCC, and inversion) in Matlab. `http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/`, 2005.
6. B. Gold, and N. Morgan. *Speech and Audio Signal Processing.* Wiley, NY, 2000.
7. M. Gruhne, K. Schmidt, and C. Dittmar. Detecting phonemes within the singing of polyphonic music. In *Proc. Int. Conference on Music Communication Science (ICOMCS)*, Dec. 5-7, Sydney/Australia, 2007.

8. T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, New Jersey, 2001.
9. K. Hornik, and D. Meyer. Consensus rankings from benchmarking experiments. In R. Decker, H. Lenz and W. Gaul (eds.) *Advances in Data Analyses*: 163–170, Springer, Heidelberg, 2007.
10. S. Krueger, M. Schafföner, M. Katz, E. Andelic, and A. Wendemuth. Speech recognition with support vector machines in a hybrid system. In *Proc. Interspeech Conference*, Lisbon/Portugal: 993-996, 2005.
11. M. Slaney. Auditory toolbox. *Apple Computer Technical Report*, 45, 1998.
12. G. Szepannek, F. Klefenz, and C. Weihs. Schallanalyse – Neuronale Repräsentation des Hörvorgangs als Basis. *Informatik Spektrum*, 28 (5):289–295, 2005.
13. G. Szepannek, B. Bischl, and C. Weihs. Towards automatic lyrics extraction from popular music. In C. Weihs, U. Ligges, A. Klapuri, R. Martin, (organizers), *Int. Workshop on Music Signal Analysis*. (presentation) Witten, Nov. 3-4, 2008.
14. G. Szepannek, T. Harczos, F. Klefenz, and C. Weihs. Combining different auditory model based feature extraction principles for feature enrichment in automatic speech recognition In *Proc. of the Specom 2009 Conference, St.Petersburg, June 21-25*: 205–210, 2009