Adiloglu, K., Annies, R., Wahlen, E., Purwins, H., Obermayer, K. (2012): A Graphical Representation and Dissimilarity Measure for Basic Everyday Sound Events, IEEE Transactions on Audio, Speech, and Language Processing, In Press

A Graphical Representation and Dissimilarity Measure for Basic Everyday Sound Events

Kamil Adiloğlu, Robert Anniés, Elio Wahlen, Hendrik Purwins, Klaus Obermayer

Abstract

Studies of Gaver [1] revealed that humans categorize everyday sounds considering the processes that have generated them: He defined these categories in a taxonomy according to the aggregate states of the involved materials (solid, liquid, gas) and the physical nature of the sound generating interaction such as deformation, friction, etc. for solids. We exemplified this taxonomy in an everyday sound database that contains recordings of basic isolated sound events of these categories.

We used a sparse method to represent and to visualize these sound events. This representation relies on a sparse decomposition of sounds into atomic filter functions in the time-frequency domain. The filter functions maximally correlated with a given sound are selected automatically to perform the decomposition. The obtained sparse point pattern depicts the skeleton of the given sound.

The visualization of these point patterns revealed that acoustically similar sounds have similar point patterns. To detect these similarities, we defined a novel dissimilarity function by considering these point patterns as 3D point graphs and applied a graph matching algorithm, which assigns the points of one sound to the points of the other sound. This novel dissimilarity measure is used in combination with a kernel machine for the classification experiments, yielding an average accuracy of $\mathbf{95}\%$ in one vs. one discrimination tasks.

Index Terms

Audio coding, audio analysis and synthesis.

This work was supported by the European CLOSED project (FP6-NEST-PATH “measuring the impossible” project no. 29085).

K. Adiloğglu and R. Anniés were with Technische Universität Berlin when the work presented in this paper was performed.

K. Adiloğlu is with INRIA, Centre Inria Rennes Bretagne Atlantique, 35042 Rennes Cedex, France (e-mail: kamil.adiloglu@inria.fr).

R. Anniés is with ARTORG Center, University of Bern, Murtenstr. 50, 3008 Bern, Switzerland (e-mail: robert.annies@artorg.unibe.ch)

K. Obermayer is with NI, Neural Information Processing Group, Technische Universität Berlin, Franklinstr. 28/29, 10587 Berlin, Germany (e-mail: oby@cs.tu-berlin.de).

E. Wahlen is a graduate student in Hamburg University of Applied Sciences. He lives in Smidstr. 5, 20535 Hamburg, Germany (e-mail: elio@elio.de).

H. Purwins is with the Music Technology Group, Department of Information and Communication Technologies, Universitat Pompeu Fabra, Roc Boronat 138, 08018 Barcelona, Spain (e-mail: hpurwins@gmail.com).

I. Introduction

Answering questions regarding recognition and perception of everyday sounds was difficult by focusing solely on musical listening or speech recognition. In her pioneering work, Vanderveer [2] defines everyday sounds as *any possible audible acoustic event which is caused by motions in the ordinary human environment.* Vanderveer studied the perception of environmental sounds in an identification experiment. She played recorded sounds to subjects and let them describe what they had heard. She observed that the subjects mostly referred to the event that caused the sound. Similarly, Houix et. al. [3] found out that the action, the objects involved and the place of the action play an important role in the recognition of these everyday sounds. Houix et al. [4] and Lemaitre et al. [5] performed a free categorization experiment in which subjects were asked to categorize kitchen sounds. The subjects also gave account whether they had categorized sounds according to their acoustic signal properties or to the event that caused them. Houix et al. and Lemaitre et al. both found out that naive listeners tend to group everyday sounds according to the events that caused them. Hence, all these studies revealed that everyday sounds are recognized and named depending on the event causing the sound. Gaver [1] called this phenomenon *everyday listening*.

Considering this phenomenon, Gaver proposed a hierarchical taxonomy of everyday sound events. He emphasized that a sound event occurs due to the interaction of two materials. Therefore, he proposed a general level in this hierarchy depending on the general categorization of these materials: *vibrating solids, liquids* and *gases*. Based on the interaction of these materials, Gaver defined the sub-categories within the solid sounds as *deformation, impact, scraping* and *rolling*. Note that these categories define very basic everyday sound events, which can happen in our daily environment, but not describe a complete environment itself. Hence, we call these kinds of sounds *basic everyday sound events*.

Recently, the computational analysis of such everyday (or more sophisticated environmental) sound events attracted increasing research interest. These studies are mainly concentrated on classification of these sounds against other sound categories like music, speech, etc. The feature vector based representation schemes dominated this field so far. Breebaart and McKinney [6] compared four different feature based approaches including spectral features (zero-crossing rate, spectral centroid etc.), Mel Frequency Cepstrum Coefficients (MFCC's), psychoacoustical features (sharpness, roughness) and auditory filterbank based (Gammatone filters) representations on classification of everyday sounds (noise), music, speech and crowd against one another. Auditory filterbank based approach outperformed the other three in average classification accuracy. Aucouturier et al. [7] used the bag-of-frames approach to model the perceptual similarity of urban soundscapes. They approximated the distribution of the MFCC's over all frames of a given sound. They compared these distributions by calculating the Kullback-Leibler divergence to detect the similarities. This method performed well on urban soundscapes but failed on polyphonic music. In general, vector based methods have difficulties in detecting events falling between two consecutive blocks. Furthermore, averaging over the blocks to generate a feature vector causes losses of certain characteristics of the given signal.

In order to overcome the disadvantages of the block based methods, Chu et al. [8] proposed a sparse approach

using Gabor atoms, applied for classification of environmental sounds via k-nearest neighbors and Gaussian mixture models. They apply matching pursuit (MP) [9] to decompose the signal into n most prominent Gabor atoms. Mean and variance of frequency, scale, and translation position of these atoms are calculated, yielding a feature vector as a representation for classification. By representing the signal by means and variances of the parameters of its atomic decomposition, the detailed spectro-temporal structure of the signal gets lost. In contrast to Gammatone functions, Gabor atoms are symmetric in time. For sounds with fast attacks and slow decays (such as many impact sounds with reverberation), a decomposition with Gabor atoms introduces an artifact prior to the attack of the signal, whereas the Gammatone function, itself with a faster attack than decay, constitutes a more suitable dictionary for this kind of sounds.

Since MP is a time consuming approach, Gribonval and Bacry [10] developed a fast version of the method by simply selecting the best sub-dictionary in an adaptive way during the coding. They used harmonic dictionaries, which the Gabor dictionaries are a special case of. Adapted to sounds with harmonic overtone spectra, as present in many musical instruments, for everyday sounds, harmonic atoms are less suitable. They tested the fast MP on note detection, which performed promisingly well depending on the values of the hyper parameters used in the algorithm. A similar idea to accelerate MP has been proposed by Coifman and Wickerhauser [11], who introduced Shannon entropy to select optimal basis functions out of a library of orthogonal wavelet-packets and localized trigonometric functions.

Apart from these classification approaches using sparse methods, sparse decomposition has been employed by Smith and Lewicki [12] on audio signals as an efficient coding scheme using biologically plausible Gammatone functions that resemble characteristics of cochlea filters. An MP scheme has been applied for finding the components, which minimize the residual error while maximizing the coding efficiency. They indicated the use of their decomposition method to depict the temporal pattern and the frequency content of the coded sounds. In order to show the efficiency of their method, they calculated the fidelity of the code in terms of the signal to noise ratio (SNR). Their studies revealed that their decomposition model is more efficient than the Fourier or wavelet representations. Furthermore, in their studies, the Gammatone functions turned out to be highly efficient for natural sounds, which include among others the everyday sounds.

We generated a basic everyday sound events database containing basic sound events conform to the taxonomy defined by Gaver. In this paper, we focused on representation, visualization, similarity and categorization of basic everyday sound events collected in this database. In order to analyze these sounds in detail, we pursue the studies of Smith and Lewicki and propose a representation scheme for sparsely coding and visualizing basic everyday sound events. We contribute to this research field by introducing a dissimilarity function based on a graph theoretical approach to calculate the similarity between those sounds, which in turn used for classifying them. This representation and dissimilarity function has been tested in a binary one-vs.-one as well as a one-vs.-all classification experiments using a supervised machine learning algorithm.

II. Basic Everyday Sound Events Database

Different from the environmental sounds, which audibly describe certain environments like streets, coffee shops, highways etc., basic everyday sound events describe materials and their elementary interactions. Therefore, compared to the environmental sounds, basic everyday sound events are recordings of a single event and not a complete environment, which possibly contain many basic everyday sound events.

We exemplified Gaver's everyday sounds taxonomy by collecting everyday sounds to generate an evaluation database. Figure 1 shows the classes and numbers of sounds per class.

We selected the sounds from the Sound Ideas [13] database. In order to reflect the diversity of the basic sound events, we selected sounds originated by significantly different sound events of the same class depending on what the Sounds Ideas database provides. It is a large library of sounds to be used mainly for film dubbing, sound design and production. Since we focus on real sound events, the subset we chose consists only of sounds that are natural recordings. We based the decisions entirely on the original work of Gaver [1]. The super and sub classes cover a broad band of basic everyday sound events in human environments, that have distinct sources and meanings.

No preprocessing has been applied to the selected sounds except for cutting 4 sec. segments of continuous sounds. The semantic descriptors of the Sound Ideas database are used for labeling the sounds. We used a hierarchical categorization with two levels. The first level consists of the three main categories identified by Gaver: *solid*, *liquid*, *gas*. The aggregate state of the object or matter that is causing and supporting the emission of a sound is the discrimination criterion.

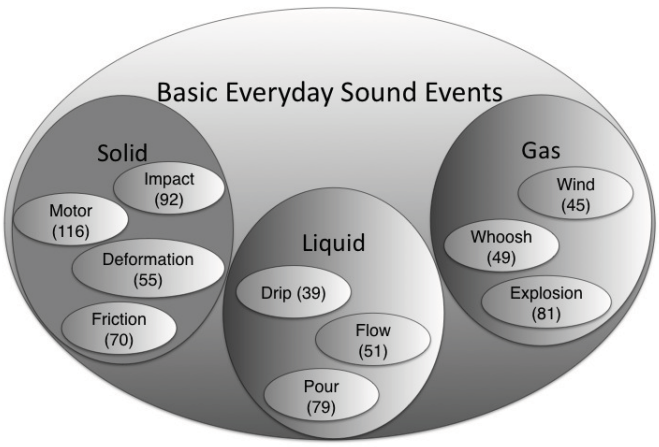


Fig. 1. The complete taxonomy of everyday sounds and the number of sounds used in the evaluation of each class.

A. Solid

Sounds emitted by the interaction of at least two *solid* objects are collected in this class. According to Gaver, the criterion for classifying a sound into this class is that the sound must be caused by mechanical contacts of one

or more solid objects. The *solid*-class consists of four sub-categories. These are *impact*, *motor*, *deformation* and *friction*.

Impact-sounds are short sounds resulting from a single instantaneous contact between two objects or parts of one object. Among others, we selected recordings of switches, single typewriter clicks, hitting sounds. The *motor*-class consists of sounds of machines and engines like of car engines in the idle state. Gaver indicates the regularity in the rolling sounds, which is the analogy to our *motor*-class. Mainly sounds of crashing objects like car crashes and glass crashes sounds constitute the *deformation* class. The sounds within the friction class involve an enforced contact of two solid objects moving against each other. Hence, the *friction* sounds category consists of recordings of squeaking and sliding doors, windows and dragging of objects, e.g. different sizes of stones.

Since the motor sounds are continuous, we cut 4 sec. segments of these sounds. The friction sounds are selected to be of any length shorter than 4 sec.

B. Liquid

Physically, liquids do not emit much sound themselves. Still there are a lot of everyday sounds that are typical for liquids and caused by water indirectly. In fact, the main sound sources are bubbles of air inside liquids that start oscillating. Alternatively, sounds emerge when liquids are reflected from or filled into solid objects making these objects oscillate. However, sounds of liquids are perceptually distinct.

The liquid sounds are split into three sub-classes: *drip*, *flow* and *pour*. The *drip*-class consists of recordings of dripping water. The examples we selected cover a wide range of drips from a countable number of drips (e.g. from a tap) to uncountable drips (e.g. drizzle). The recordings also differ mainly in the reverberation of the room and the material type (liquid itself or solid), where the drips are reflected or absorbed, i.e. at the point of sound emission. Hence the distinguishing factor of the *drip*-class to the other *liquid*-classes according to the semantics of the material interaction is that small distinguishable portions of water fall onto a surface. The *flow*-class mainly encapsulates movements of large portions of water that create swirls of air causing sound. Hence, it consists of examples of running water taps, waves on the shore, different sizes of rivers from brooks up to large rivers and similar movements of water. Depending on the amount of water, high background noise is audible within these recordings, e.g. very large waves or a very large river. The third *liquid*-class is called *pour*. In there, we collected sound recordings of the interaction of transporting a portion of water from a vessel A to a vessel B through air, for instance filling a glass from a bottle. Sound is emitted by air bubbles that appear during the action. This definition seems close to that of the *flow*-class, but we have selected only sounds, where there is a clear directed relocation of the water or drink from A to B. Because there are also sparkling drinks involved, the sounds are a mixture of the pouring action and the sounds that comes from the carbonated drink itself.

We cut 4 sec. segments of these sounds, because liquid sounds are continuous.

C. Gas

Aerodynamic sounds sources are more direct. These sources create the sound by changing the atmospheric pressure. This can happen suddenly or as a steady process. The former are explosive sounds that populate the *explosion*-class. Gun shots and larger detonations from TNT are possible candidates for this sub-class proposed by Gaver. Explosions result in a very energetic short bang, that is almost a Dirac impulse.

As *wind* we considered sounds resulting from a constant movement of air: recordings of wind sound at different locations, steam, and blow sounds. Similar to the motor and liquid sounds, we cut 4 sec. segments of the wind sounds.

The third class is labeled as *whoosh*. The sound events of this class are caused by objects passing by the listener (microphone) with a high velocity. Arrow sounds, airplanes and very fast cars passing by the listener constitute three main sound groups represented within this class. Furthermore, we included transient air blows (e.g. flame thrower) into the whoosh class as well to increase the diversity of the class. Since the flight duration of an arrow or an airplane or the time needed until a fast car approaching and passing by the listener differ completely, the durations of the sounds within this class range from half a second up to half a minute.

III. Representation of Everyday Sound Events

In the present paper, we utilise a sparse representation scheme used for visualization of everyday sound events as well as in a classification scenario to categorize them. In the following sections, we will explain how to decompose a sound into a sparse set of filter functions. Following the terminology of Smith and Lewicki, each of these functions will be called a *spike*, and the whole set of spikes will be called a *spike code*.

In order to construct the spike code, we prefer the biologically motivated Gammatone filterbank as the set of basis functions. A Gammatone filterbank [14] of M filters with center frequencies f_m and bandwidths b_{f_m} is defined by the Gammatone function with a temporal offset t^* as follows:

$$\gamma_{f_m,t^*}(t) = (t-t^*)^3 \cos(2\pi f_m) e^{-2\pi b_{f_m}(t-t^*)} \tag{1}$$

with $1 \leq m \leq M$. For $t^* = 0$, this function indicates the impulse response of the Gammatone filter. The relationship between the bandwidths and the center frequencies of the filters within the filterbank are determined according to the ERB scale, proposed by Glasberg and Moore [15]. They indicate that the bandwidth of the filter corresponds to a fixed distance on the basilar membrane. The biological plausibility of the Gammatone filterbank as a model of the basilar membrane is also supported by the high similarity of the impulse response of a Gammatone filter to the impulse response of the basilar membrane measured in cats [16]. Besides, the magnitude characteristics of a gammatone filter is similar to the magnitude characteristics of the rounded exponential filter, which is used for representing the human auditory filter [17]. Hence, the use of a Gammatone filterbank for coding sounds accounts for the basilar membrane side of the auditory perception.

We use the implementation of the Gammatone filterbank in the auditory toolbox by Malcolm Slaney [18], [19].

A. Sparse Signal Representation

In a sparse, shiftable representation method based on atom-like filter functions, a sound signal $x(t)$ can be approximated as a linear combination of K Gammatone functions $\gamma_{f_k,t_k}(t)$,, selected from a unit-normed Gammatone filterbank of M filters, with coefficients (amplitudes) a_k and residual $\epsilon_{K+1}(t)$:

$$x(t) = \sum_{k=1}^{K} a_k \gamma_{f_k,t_k}(t) + \epsilon_{K+1}(t). \tag{2}$$

Each selected Gammatone function, called a spike s_k, is composed of the temporal offset t_k, the center frequency f_k of the Gammatone filter, and its amplitude a_k. We employ matching pursuit [9], [12] to iteratively determine the spikes s_k and to minimize the residual. The k^{th} spike, $1 \leq k \leq K$, with the time offset t_k and the center frequency f_k is selected to be the Gammatone filter $\gamma_{f_m,t*}(t)$, $1 \leq m \leq M$, which maximally correlates with the signal $\epsilon_k(t)$:

$$(f_k, t_k) = \text{argmax}_{f_m,t^*} < \epsilon_k(t), \gamma_{f_m,t^*}(t) > . \tag{3}$$

In his auditory toolbox, Slaney [18], [19] used fixed window lengths for each filter in the filterbank. However, the energy levels decrease much slower for the low frequency filters than for the high frequency filters. Therefore, we adapted the window lengths considering the center frequencies. We calculated the positions for each filter within the filterbank, where the total energy in the time envelope falls to its thousandth. We used these position values in time as the window lengths to calculate the scalar product.

For each filter, we traversed the sound signal using a hop size and the scalar product between the signal and the filter is calculated in each iteration. The filter with the center frequency f_m at a certain position within the signal $t*$ yielding the highest scalar product is selected to be the k^{th} spike $s_k = \gamma_{m_{f,t^*}}$. The amplitude a_k of the spike s_k is defined to be the scalar product between the corresponding filter and the residual signal, as shown in (3).

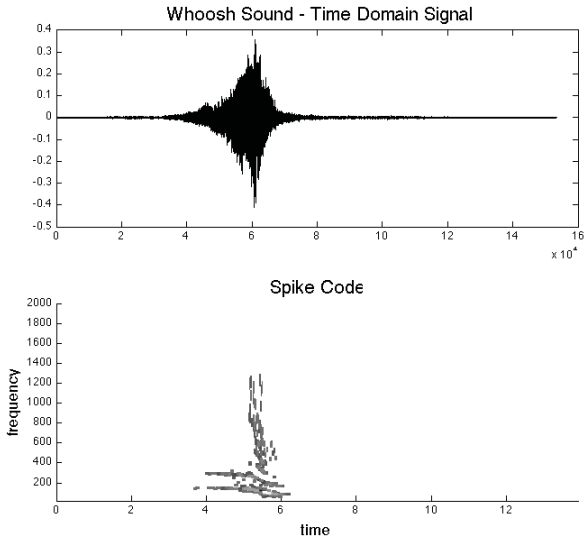
After finding the optimal values t_k, f_k and a_k for the k^{th} spike s_k, we update the residual by

$$\epsilon_{k+1}(t) = \epsilon_k(t) - a_k \gamma_{f_k,t_k}(t) \tag{4}$$

Finally, for $k = K$ we yield (2). By varying K, the sparsity of the representation can be traded off versus its SNR. Increasing K increases the SNR of the representation.

Figure 2 shows the wave form and the spike code of a *whoosh* sound. We observe easily, how the spike code captures the skeleton of the sound. Note that salient areas in the spectrogram are coded with more spikes than other areas, whereas no spikes have been used to encode the rest of the signal. Most importantly, one can easily recognize these salient areas within the sound and match them directly to the spike code. For instance, in this figure, we have a recording of a fast car passing by. Therefore the spikes are concentrated in the middle of the sound, where the car reaches the person (or microphone) and passes by. The algorithm did not spend any spikes when the car is far from the person. Hence, the spike code is an easy to understand method for visualizing sounds properly.

Figure 3 shows spike codes of one sample sound from each class of our basic everyday sounds database. As we mentioned for Figure 2, by considering not single spikes but the groups of spikes within the whole code, one can



(a)

Fig. 2. For a sound from the main class *gas*, subclass *whoosh*, the sound wave (top) and the spike code (bottom) generated by a Gammatone filterbank of $M = 256$ filters and $K = 256$ spikes. spikes.

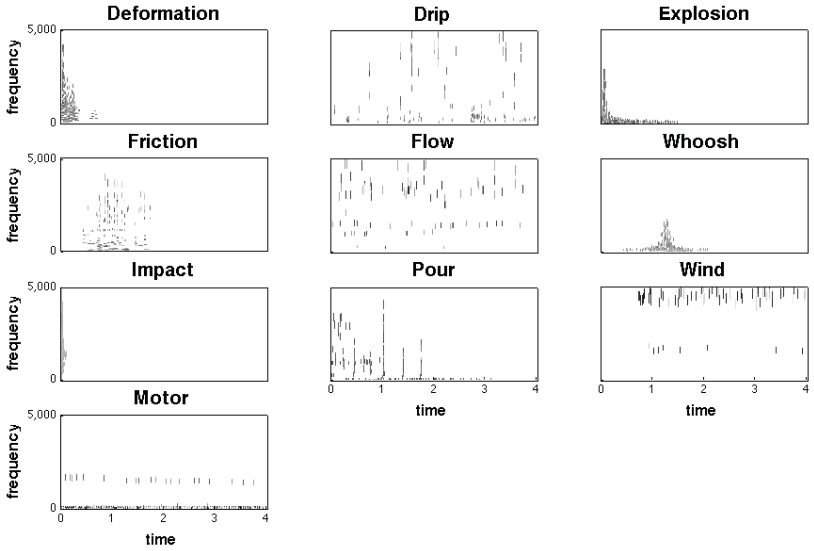


Fig. 3. Spike codes of one sample sound from each class of the basic everyday sound events database, generated by using a Gammatone filterbank of $M = 256$ filters and $K = 256$ spikes.

easily assign these groups to the salient events within the coded sound. Furthermore, considering the spike code as a whole reveals that spike codes indicate common patterns for sounds with common attributes. Hence, spike codes of sounds having similar auditory features, e.g. recordings of similar events like sliding doors and windows etc. and hence being from the same class of a sound database, show similar patterns, whereas spike codes of sounds of different classes are significantly different. Apart from this fact, the spike codes can be regarded as a representation to perform classification of these sounds by making use of these similarities between them. In the following section, we explain in detail, how we captured these similarities. However, beforehand, we would like to mention three state of the art representation schemes, which we have used as comparison.

B. Other Representation Schemes

MFCC's [20] are a well established representation scheme which dominates applications in speech recognition and music processing. The frequency bands in this representation are spaced equally on the Mel scale, which is a perceptually relevant scale indicating the auditory response on the basilar membrane. We use the first 13 MFCC coefficients. A feature vector is computed by taking the mean, variances, finite differences between consecutive MFCC coefficients and the variances of these differences over all frames, adding up to one 52-dimensional vector for a sound example.

As a comparison, we use a feature set we will call Low-level signal features (SLL). It consists of the energy, zero crossing rate, spectral centroid, roll-off frequency and their variances, finite differences and the variances of the differences giving a 16-dimensional feature vector, similar to the above feature construction.

For a further comparison, a set of timbre descriptors is used: perceptual spectral centroid, relative specific loudness, sharpness, roughness, signal autocorrelation, zero crossing rate, time frame, log attack time, temporal increase, decrease and centroid, effective duration, energy modulation frequency, energy modulation amplitude. As an implementation we use the IRCAM descriptor [21].

IV. Classification of Sounds

The similarities between spike codes can be captured without destroying the original structure of the pattern. In this paper, we propose a novel, structure preserving dissimilarity function to calculate the dissimilarity between two spike coded sounds in two steps. In the first step, the dissimilarity between two spikes of different spike codes is calculated. In the following step, the total dissimilarity between the given spike codes is calculated by taking the minimum of the sums of spike dissimilarities of all pairwise assignments. Intuitively, this method measures the minimal effort to transform one spike code into the other in terms of the single spike dissimilarity (see Figure 4).

A. Dissimilarity between Two Spikes

The dissimilarity $d_s(s, s')$ between two spikes $s = (t, f, a)$ and $s' = (t', f', a')$ is composed of three individual dissimilarities, namely the dissimilarity $d_t(t, t')$ between times offsets, $d_f(f, f')$ between center frequencies, and

$d_a(a, a')$ between amplitudes:

$$d_s(s, s') = \tau d_t(t, t') + \phi d_f(f, f') + (1 - \tau - \phi) d_a(a, a'), \tag{5}$$

with $\tau, \phi \geq 0, \tau + \phi \leq 1$. Parameters τ and ϕ allow to emphasize either the temporal, spectral, or amplitude aspect, while guaranteeing that the weights sum up to 1.

In order to calculate the temporal dissimilarity, for each sound the spike with minimal time offset $t_{\min}$ is determined. This is used to omit the "silence" before the first spike when comparing two sounds. $t_{\max}$ be the maximal sound length across all sounds measured between the first and the last spike of each sound. For time offsets t, t' of two spikes the time dissimilarity holds:

$$d_t(t, t') = \frac{((t - t_{\min}) - (t' - t'_{\min}))^2}{t^2_{\max}}. \tag{6}$$

For calculating the frequency dissimilarity, we consider logarithmic frequencies, according to the Weber-Fechner law. However beforehand, we normalize the frequencies with the highest center frequency $f_{\max}$ of the filters within the filterbank. For frequencies f, f' of two spikes we define the dissimilarity to be:

$$d_f(f, f') = (\frac{\log f - \log f'}{\log f_{\max}})^2. \tag{7}$$

For the amplitude dissimilarities, we take the absolute values of the amplitude, which normally could be negative as well. In the calculations, we divide the amplitude a by the maximum amplitude $a_{\max}$ of the sound, thereby equalizing the volume. The maximum amplitudes are determined by first taking the absolute values of the amplitudes and choosing the amplitude with the largest value. Hence, the dissimilarity is defined as

$$d_a(a, a') = (\frac{\log |a|}{\log a_{\max}} - \frac{\log |a'|}{\log a'_{\max}})^2. \tag{8}$$

These separate dissimilarities between amplitude, frequency and time offsets are summed up in (5) to calculate the total spike dissimilarity. After calculating the dissimilarity between two spikes, the dissimilarity between two spike-coded sounds can be achieved.

B. Dissimilarity between Spike-Coded Sounds

For two sounds, we consider their spike codes $\mathbf{s} = \{s_1, \ldots, s_K\}$ and $\mathbf{s}' = \{s'_1, \ldots, s'_K\}$. Given the dissimilarity function $d_s(s_i, s'_j)$ for two spikes $s_i \in \mathbf{s}, s'_j \in \mathbf{s}'$, we can define a dissimilarity $d(\mathbf{s}, \mathbf{s}')$ between two sounds by establishing a bijection between $\mathbf{s}$ and $\mathbf{s}'$. For a permutation $\mu \in S_K$ we assign to each spike in $\mathbf{s}$ exactly one spike in $\mathbf{s}'$, so that $s_i \to s'_{\mu(i)}$. Then we define the dissimilarity between two spike-coded sounds as the scaled sum of the dissimilarities between the K corresponding spikes:

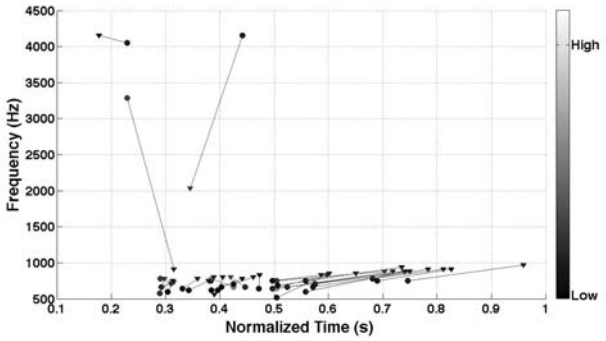
$$d(\mathbf{s}, \mathbf{s}') = \frac{1}{K} \sum_{i=1}^{K} d_s(s_i, s'_{\mu(i)}), \tag{9}$$

with μ minimizing the dissimilarity:

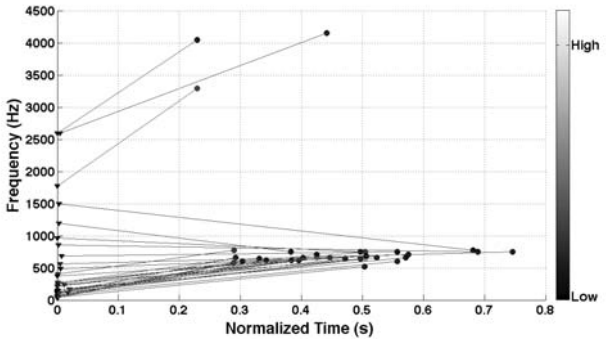
$$\mu = \operatorname{argmin}_{\mu \in S_K} d(\mathbf{s}, \mathbf{s}'). \tag{10}$$

C. Graph Matching

Instead of solving the problem of minimizing the assignment μ in (10) directly, we consider the spike codes of two given sounds as two point graphs combined in a bipartite graph. In this constellation, the vertices are the spikes $s_i \in \mathbf{s}, s'_j \in \mathbf{s}'$ of two spike-coded sounds $\mathbf{s}, \mathbf{s}'$, where each sound corresponds to a disjoint subgraph of the bipartite graph, and the weights (similarities) between them are derived from the dissimilarity d_s : $w_{ij} = 1 - d_s(s_i, s'_j)$ with $\mu(i) = j$. This consideration converts the problem into a combinatorial one of finding an optimal matching of the weights in a bipartite graph, optimal in terms of the minimum dissimilarity defined in (10). For this purpose, we used the Hungarian algorithm [22]. We consider a matching to be a subset of the edges of the given graph containing each vertex only once. In a perfect matching, every vertex within the graph is adjacent to an edge. The Kuhn-Munkres Theorem [22] guarantees the convergence of the algorithm to a perfect matching.



(a) Pour 1 ('o') - Pour 2 ('v')



(b) Pour ('o') - Explosion ('v')

Fig. 4. Spike pattern comparison: a) two sounds from the *pour*-class, b) a *pour* and an *explosion* sound.

Figure 4 shows two examples of matching results performed by the Hungarian algorithm. The upper plot indicates the matching results between two sounds of the same sound class. The lower plot, on the other hand, depicts matching results of the sounds coming from two different classes. These two plots exemplify that the dissimilarities between spike patterns of different sound classes are in general larger than the dissimilarities among sounds of the same class.

Let $\mathcal{S}_K$ be a set of sounds, encoded by K spikes. The dissimilarity $d(\mathbf{s}, \mathbf{s}') : \mathcal{S}_K \times \mathcal{S}_K \rightarrow \mathbb{R}^+$ is symmetric and it holds $d(\mathbf{s}, \mathbf{s}') = 0$ for $\mathbf{s} = \mathbf{s}'$. For L sounds $\mathbf{s_i}, \mathbf{s_j} \in \mathcal{S}_K$ $(1 \leq i, j \leq L)$, $d(\mathbf{s_i}, \mathbf{s_j})$ gives $L \cdot L$ pairwise dissimilarities, expressed as the dissimilarity matrix:

$$\mathbf{D} = (d(\mathbf{s_i}, \mathbf{s_j}))_{1 \leq i,j \leq L}. \tag{11}$$

D. Distance Substitution Kernel

In most of the real classification problems, data is not linearly separable. A common approach to overcome this difficulty is the use of a distance substitution kernel [23].

A given dissimilarity function $d(\mathbf{s}, \mathbf{s}')$ is embedded in a (high-dimensional) space $\mathcal{H}$ using a kernel function $\mathbf{K}$ so that in $\mathcal{H}$ we can separate the classes linearly by hyperplanes.

Given spike dissimilarity function d in (9), we use a Gaussian kernel to define the so-called $L \times L$ Gram matrix:

$$\mathbf{K} = (\exp \frac{-d(\mathbf{s_i}, \mathbf{s'_j})}{2\sigma^2})_{1 \leq i,j \leq L}. \tag{12}$$

σ^2 is the variance (sometimes called length scale) parameter which has to be determined. In the state-of-the-art applications, the Gram matrices should be positive semi-definite that normally arises from norms in the feature space. However, the Gram matrix $\mathbf{K}$ of the proposed dissimilarity function is not positive semi-definite.

Note that our dissimilarity function is not exactly an Euclidean distance. Furthermore, considering the spikes as axes in a K dimensional system, the performed matching can be interpreted as a switch of axes in this system. This means that the algorithm matches a particular spike s_k of a given sound $\mathbf{s}$ to different spikes in different sounds. Therefore, the dissimilarity function defined for spike coded sounds does not define an Euclidean space.

Let us explain the situation on an example. Given sounds $\mathbf{s}, \mathbf{s}', \mathbf{s}$", the dissimilarity between the sounds $\mathbf{s}$ and $\mathbf{s}'$ for the i-th spike of sound $\mathbf{s}$ is defined as $d_s(s_i, s'_{\mu(i)})$. Similarly, the distance between the sounds $\mathbf{s}$ and $\mathbf{s}$" for the same spike is defined as $d_s(s_i, s''_{\nu(i)})$. However, the matching $\mu(i)$ is not equal to $\nu(i)$.

Therefore, the Gram matrix defined in (12) does not constitute a valid kernel. In order to be able use this Gram matrix in a kernel machine, either one has to transform $\mathbf{K}$ such that it becomes positive semi-definite (see Graepel et al. [24]) or one has to resort to kernel machines that do not require positive semi-definite kernels. We prefer the latter approach.

There are several approaches to overcome the necessity of a positive semi-definite Gram matrix. Balcan et al. [25] proposed a theory of learning with similarity functions not necessarily positive semi-definite. They showed that for pairwise similarity functions satisfying certain conditions (conform to a certain definition), there is an explicit transformation of the data, after which a standard large-margin classifier can be applied. In another study, Hochreiter et al. [26] [27] proposed a variant of the standard SVM called the Potential SVM (P-SVM), which selects models using the principle of structural risk minimization. In contrast to the standard SVM approaches, the P-SVM is based on another objective function and another set of constraints, which lead to an expansion of the classification or regression function in terms of support features. The optimization problem is quadratic, always well defined, suited for dyadic data, and neither requires square nor positive semi-definite Gram matrices. Therefore, the

P-SVM approach can be used without any pre- or post-processing of the measured as well as constructed Gram matrices using an indefinite kernel function. Hence, in this study, the spike kernel is used in combination with the P-SVM to perform classification.

V. Experiments

Both the matching pursuit algorithm and the Hungarian algorithm are time consuming operations. Coding the sounds with too many spikes takes a large amount of time, which in return does not necessarily yield either a better visualization or a better classification accuracy. Therefore, it is essential to determine the optimal number of spikes. However, it is essential to span the whole frequency space. Smith and Lewicki found out that with $M = 64$ filter functions, an efficient time-relative code is possible. Increasing the number increases the efficiency linearly, but time needed to compute increases exponentially. In our experiments, we used $M = 256$ filters, which ensures highly efficient codes according to their results.

In order to find the optimal number of spikes needed for the coding in terms of the classification accuracy, we performed classification experiments with different number of spikes for randomly selected two classes – *drip* and *flow*. We coded the sounds with $K = 32, 64, 96, \ldots, 256$ spikes and calculated the balanced test accuracies and the duration needed for these experiments. The obtained accuracies shown on the left hand side in Figure 5 indicate an increase as the number of spikes increases. However there is a plateau between $K = 160$ and 224 spikes. The time needed for these experiments – shown on the right hand side in the same figure – increases exponentially as the number of spikes increases. Hence, for the sake of computational costs – the Hungarian algorithm has complexity of $O(K^3)$ – we ran the experiments with $K = 192$ spikes, which yields an acceptable accuracy in acceptable computation time.

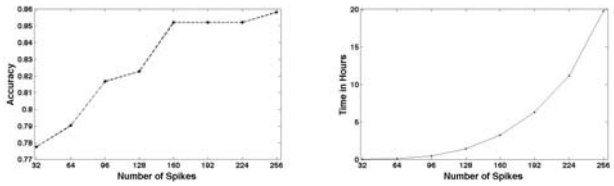


Fig. 5. Left: The balanced test accuracies for the classification experiments of the sound classes *drip* and *flow* as a function of number of spikes used for coding the sounds. Right: The time needed to perform these experiments.

To overcome the disadvantages of block based methods, we used a hop size of one sample during the coding to find the best fit. We found that using hop size values larger than 10 samples, the difference in the quality between the original and the resynthesized sound is clearly audible.

In order to account for the time course and the spectral content of the sounds, we set $\tau = 0.4$ for the temporal dissimilarity d_t (6) and $\phi = 0.4$ for the frequency dissimilarity d_f (7). The classification experiments performed with these weight values among others yielded the best results in terms of accuracy.

In order to evaluate the accuracies of the dissimilarity matrices, we have performed classification experiments with the P-SVM to discriminate one class from another, for all pairs of classes separately, 45 pairs in total. Furthermore, we have performed detection experiments, again with P-SVM, where we measured the detection accuracies of one class vs. all other classes, for all classes separately, 10 experiments in total. The hyperparameter ϵ of the P-SVM and the kernel size γ were varied in a grid search: $0.1 \leq \epsilon \leq 1.0$, $2^{-5} \leq \gamma \leq 2^{10}$ to find an optimal setting.

We measured the accuracies of all these experiments by using the leave-one-out cross validation method.

VI. Classification Results

This sound database corpus has been classified by using the spike coding method as well as by using the MFCC, SLL, and TIMBRE descriptors for comparison.

Figure 6 shows the accuracies for binary discrimination tasks between all pairs of classes in an upper triangular form as well as the one vs. the rest detection accuracies underneath. The one vs. the rest detection experiments have been performed only for the best three representation methods. As it can be seen in the upper part of the figure, the TIMBRE descriptors perform significantly worse than the other three for the one vs. one discrimination tasks. Therefore we excluded them from the one vs. the rest detection experiments.

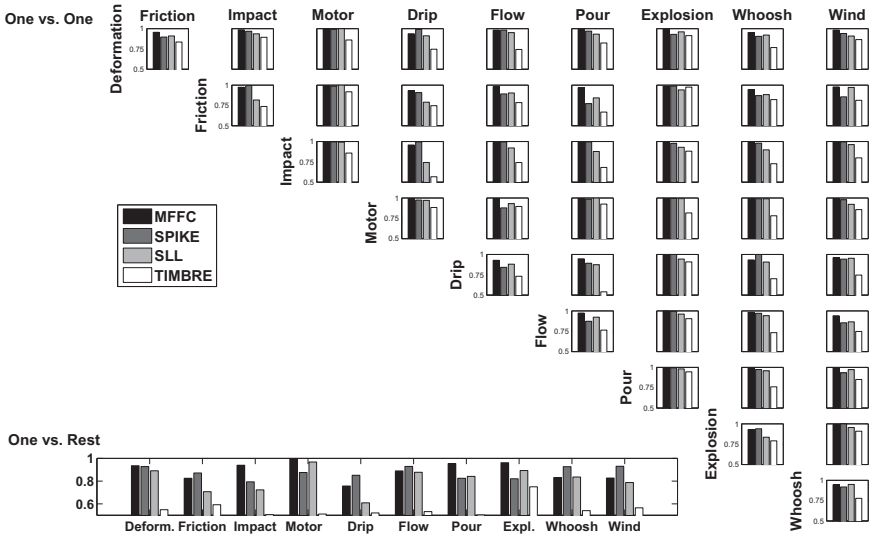


Fig. 6. Binary discrimination accuracies of four different representation schemes (the MFCCs, the spike code, the SLLs and the timbre descriptors)in the upper part and one vs. the rest detection accuracies of the first three representation methods in the lower part of the figure.

The results of the one vs. one discrimination tasks indicate that the overall performance of the MFCC representation outperforms the other three representations, including the spike representation. However, the overall accuracy obtained by the spike representation is very close to the MFCC accuracy. Furthermore, for several particular cases, for friction vs. impact for instance, the spike representation outperforms the MFCC representation. SLL and TIMBRE

TABLE I

THE BALANCED PREDICTION ACCURACIES OF THE BINARY ONE VS REST DISCRIMINATION EXPERIMENTS FOR THE MOTOR CLASS OBTAINED BY THE GENERAL NORMALIZATION SCHEME AND ANOTHER NORMALIZATION SCHEME PROPOSED FOR THE MOTOR CLASS.

	Defr	Fric	Impact	Motor	Drip	Flow	Pour	Expl	Who	Wind
Special	0.9915	0.9892	1		0.9957	0.9833	1	1	0.9915	0.9957
General	0.9909	0.9857	0.9945		0.9743	0.8780	0.9851	0.9938	0.9900	0.9782

show mixed results. Only SLL has a fairly good performance in the one vs. one task. A high variance and/or poor accuracy of SLL and TIMBRE in the other cases make them less applicable here.

In the one vs. the rest detection experiments, the spike code outperforms the other representation schemes including the MFCCs in five particular cases. In the other five cases, MFCCs perform the best.

TABLE II

MEAN AND STANDARD DEVIATIONS OF THE RESULTS

		MFCC	SPIKE	SLL	TIMBRE
one vs. one	μ	97.70	94.86	92.44	80.29
	σ	2.4	5.5	5.7	9.4
one vs. rest	μ	89.19	87.51	81.32	55.68
	σ	7.8	5.2	10.71	7.4

Table II shows the average discrimination and detection performances of the model for the results in Figure 6. In the average accuracy, the MFCCs perform slightly better than the spike representation. However, their standard deviation is larger as well, meaning that their uncertainty about the average accuracies is higher compared to the accuracies of the spike representation.

For the spike representation, one should also consider that we employed the same normalization method and the same coefficient values in all dissimilarity calculations. We wanted to demonstrate the overall performance of the method without embedding specific information about the sounds into the dissimilarity calculations. However, our dissimilarity function enables to embed problem specific expert knowledge into the calculations by changing the normalization and the values of the coefficients. For instance, while the impact sounds are all short pulses, the motor sounds are all continuous and the length of the whoosh sounds are quite variable. Hence, incorporating a suitable normalization in combination with coefficient values accounting for the correct relationship between the three dimensions of the spikes, the classification accuracies can be improved. We demonstrate this possibility on the motor class by normalizing the times, amplitudes and frequencies locally within each sound. Table I shows the discrimination accuracies of the experiments performed with (Special) and without (General) this normalization for the motor class. As Figure 3 also indicates, the motor sounds depict a very regular pattern along all three

dimensions. For this particular example they have mainly two frequency components, one very low component and one component around 1800 Hz. From sound to sound, these frequency components can change, but regularity of the sound and in turn the spike pattern does not change. By normalizing the frequency dimensions locally for each sound, these frequency differences can be removed. Hence the regularity of the spike patterns within the motor class can be detected easily. The results shown in Table I indicate a clear improvement compared to the results obtained with the general normalization method proposed in Section IV-A. Similarly, this kind of prior knowledge can be embedded for other cases as well to improve the results.

VII. Discussion

1) Noise: The amount of noise in basic everyday sound events can not be neglected in the analysis and classification. How is noise encoded with the gammatone spike approach? Firstly, the encoding can be understood as a denoising step that emphasizes the contours within the sound which are perceptually important and result in a robust dissimilarity function. Secondly, from a certain threshold of spike numbers the noise in the signal is encoded in a probabilistic way. The position of a Gammatone atom in the noise is determined by the noise distribution. Encoding sounds with an inherent portion of noise, like streaming water, could emphasize contours in the sound which are perceptually non-existent (phantom spikes). With psycho-acoustic experiments the number of spikes threshold from which a human cannot distinguish between a set of single phantom spikes and noise could be measured and used to avoid this effect.

2) Comparison to other Filterbanks: We decompose sounds as a linear combination of Gammatone functions as atoms. In contrast to the Short-Term Fourier Transform (STFT) with constant bin widths, we use filter bandwidths according to the perceptual ERB scale that increase exponentially with frequency. The Mel frequency scale used by MFCC's is spaced in a way similar to the ERB scale. Both the STFT and the MFCC's are calculated with a constant frame length in the temporal domain. For our approach, the length of the temporal filter decreases and the bandwidths of the filters increase with increasing center frequency and vice versa. Instead of plain sine functions, the atoms in our decomposition are biologically plausible Gammatone functions, similar to Solbach et al.'s [28] wavelet filterbank built from Gammatone filters.

3) Relation to Other Matching Methods: Let us discuss our approach in relation to three other matching methods, namely the Bag-of-Frames Distance, the Wasserstein-Mallows Earth Mover Distance, and Dynamic Time Warping.

Usually, the Bag-of-Frames Distance (BoF) [7] does not consider time as a dimension explicitly encoded in the representation. Therefore, a sound is represented in the same way as the same sound played backwards, even though these are perceptually very different. They would yield a high dissimilarity in our measure. From our sparse representation of a sound, it would be more difficult to determine the probability density than it would be using all full-dimensional feature vectors like it is done in the BoF Distance. The BoF Distance does not explicitly consider the difference between two features. It just gives a density estimate, thereby measuring if certain values happen equally often.

The method presented in this paper can be considered as an approximation of the Wasserstein-Mallows Earth

Mover Distance (EMD) [29]). The Hungarian algorithm we used here is a special case of EMD, where the optimal flow matrix is a 0-1 permutation matrix, i.e. the earth is only moved between two points. A binary value is assigned to the mass if a triple of frequency, time and amplitude is considered to be among the selected n most prominent spikes (mass 1) or not (mass 0). A variant would be to consider the amplitude as the mass in the EMD. Due to the sparsity of our representation of the sound and the 1-to-1 mapping between spikes (contrary to a soft-assignment proposed in the EMD) the computational expense of our method is significantly reduced in comparison to EMD.

Dynamic Time Warping (DTW) [30] is another alignment method aligning two time series. In order to align two sounds, DTW maps the temporal evolution of one sound onto the time course of the other sound. In contrast to our dissimilarity approach, which relies on the three-dimensional structure of the spike code, considered as a point graph, DTW is a non-linear but monotone acting along the time axis by matching several time steps (samples) of one sound to one single time in the other sound. In order to perform this alignment, the spike code should be converted into a two-dimensional list. Hence, our dissimilarity function preserves the three-dimensional structure that gets lost when applying DTW.

VIII. Conclusion

Following the everyday listening phenomenon, we realized Gaver's taxonomy as a database of basic everyday sound events on two levels of description, the sound generating materials on the top level and the interactions between them on the bottom level. We exemplified these sound categories with recordings of isolated sound events taken from the Sound Ideas database.

We coded the basic everyday sound events within this database by incorporating a sparse representation scheme using the Gammatone filters. Previous studies have shown that these filter functions are suitable for coding everyday sounds. The use of the Gammatone filters also links the representation to biological and psychoacoustic findings. In order to find the positions of the filter functions, we utilized the matching pursuit algorithm. This yielded an efficient code for these sounds, which accounts not only for spectral and energy properties, but also for temporal characteristics. The efficiency of the sparse representations is better than the Fourier or wavelet representations [12], but using other encoding algorithms could even increase these values.

Plotting the spike code yields a visualization, a salient skeleton or a cartoon of the sound. Such a transcription provides valuable information about the sound visually indicating the audible similarities and differences between mostly and / or partly similar sounds, e.g. sounds stemming from the same sound class.

The audible similarities and differences easily observable on spike codes inspired us to define a structure preserving dissimilarity function incorporating the graph theory. Thereby, we extended sound classification from Euclidean distances between feature vectors to the more general scenario of non-metrical dissimilarities. We employed a distance substitution kernel and utilized the P-SVM to perform the classification. With the P-SVM, we introduced an analysis method in sound and music computing that can handle any dissimilarity measure without any requirement to a positive semi-definite kernel structure. The potential of the method has been demonstrated by predicting the attributes of basic everyday sound events. In binary discrimination and detection scenarios, our

method performed promisingly well.

The graphical approach, we applied to define the dissimilarity function for sparse representations is, to our knowledge, a new paradigm in the music and sound processing domain for classification purposes. The promising initial results presented in this paper indicate the potential of this way of thinking. Even though the average classification results of the MFCCs are slightly better than the ones obtained by using our dissimilarity function, the flexibility of our dissimilarity function enables to embed some problem specific knowledge easily, so that the accuracy of the results increases. The improvement in the results of the motor class indicates this fact, which is not possible for several other methods including the MFCCs. Therefore, the authors are strongly convinced that this method is applicable to a wide variety of sounds.

In a further study, a prototypical spike code will be generated for a given sound class in an unsupervised way, by adapting the k-means algorithm to the proposed dissimilarity measure.

Such a prototypical spike code can reveal interesting features common within the given sound class. These features can be visualized by the method and yet since a spike code can be re-synthesized, the prototypes become audible as well. An audio-visually available, prototypical sound pattern generated from a set of real examples raises many new research questions, concerning not only the unsupervised categorization of sounds but also common perceptual features as well as their design and optimization.

ACKNOWLEDGMENT

We would like to thank Olivier Houix for his help with the sound selection and Yon Visell for inspiring discussion.

REFERENCES

[1] W. W. Gaver, "How do we hear in the world? Explorations in ecological acoustics," *Ecological Psychology*, vol. 5, no. 4, pp. 285–313, 1993.

[2] N. J. Vanderveer, "Ecological acoustics: human perception of environmental sounds," PhD thesis, Cornell University, 1979.

[3] O. Houix, G. Lemaitre, N. Misdariis, P. Susini, K. Franinovic, D. Hug, J. Otten, J. Scott, Y. Visell, D. Devallez, F. Fontana, S. Papetti, P. Polotti, and D. Rocchesso, "Everyday sound classification. part 1 : State of the art," Commission Européenne, Tech. Rep., 2003.

[4] O. Houix, G. Lemaitre, N. Misdariis, and P. Susini, "Classification of everyday sounds: Influence of the degree of sound source identification," *The Journal of the Acoustical Society of America*, vol. 123, no. 5, 2008.

[5] G. Lemaitre, O. Houix, N. Misdariis, and P. Susini, "Naïve and expert listeners use different strategies to categorize everyday sounds," *The Journal of the Acoustical Society of America*, vol. 123, no. 5, 2008.

[6] J. Breebaart and M. McKinney, "Features for audio and music classification," in *Proceedings of the International Conference on Music Information Retrieval*, Baltimore, 2003.

[7] J.-J. Aucouturier, B. Defreville, and F. Pachet, "The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music," *Journal of the Acoustical Society of America*, vol. 122, no. 2, pp. 881–891, 2007.

[8] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental sound recognition using MP-based features," in *Proceedings of ICASSP*, 2008.

[9] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[10] R. Gribonval and E. Bacry, "Harmonic decomposition of audio signals with matching pursuit," *IEEE Transactions on Signal Processing*, vol. 51, no. 1, pp. 101–111, 2003.

[11] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Transactions on Information Theory*, vol. 38, pp. 713–718, 1992.

[12] E. Smith and M. S. Lewicki, "Efficient coding of time-relative structure using spikes," *Neural Computation*, vol. 17, pp. 19–45, 2005.

[13] "Sound ideas sound database, http://www.sound-ideas.com." [Online]. Available: http://www.sound-ideas.com

[14] R. D. Patterson and J. Holdsworth, "A functional model of neural activity patterns and auditory images," *Advances in Speech, Hearing and Language Processing*, vol. 3, pp. 547–563, 1996.

[15] B. R. Glasberg and B. C. J. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing Research*, vol. 47, pp. 103–138, 1990.

[16] L. H. Carney and C. T. Yin, "Temporal coding of resonances by low-frequency auditory nerve fibers: Single fibre responses and a population model," *J. Neurophysiology*, vol. 60, pp. 1653–1677, 1988.

[17] R. D. Patterson and B. Moore, "Auditory filters and excitation patterns as representations of frequency resolution," in *Advances in Speech, Hearing and Language Processing*, B. Moore, Ed. London: Academic Press Limited, 1986, pp. 123–177.

[18] M. Slaney, *A matlab toolbox for auditory modeling work*, Interval Research Corporation, 1998.

[19] ——, *An Eifficient Implementation of the Patterson-Holdsworth Auditory Filter Bank*, Apple Computer, 1993.

[20] B. Logan, "Mel frequency cepstral coefficients for music modeling," in *International Symposium on Music Information Retrieval*, 2000.

[21] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the CUIDADO project," IRCAM, Analysis/Syntesis Team, Tech. Rep., 2004.

[22] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1955.

[23] B. Haasdonk and C. Bahlmann, "Learning with distance substitution kernels," in *Proc. 26th DAGM Symp.*, 2004, pp. 220–227.

[24] T. Graepel, R. Herbrich, P. Bollmann-Sdorra, and K. Obermayer, "Classification on pairwise proximity data," in *Advances in Neural Information Processing Systems*, 1998, pp. 438–444.

[25] M.-F. Balcan, A. Blum, and N. Srebro, "A theory of learning with similarity functions," *Machine Learning Journal*, vol. 72, no. 1-2, pp. 89–112, 2008.

[26] S. Hochreiter and K. Obermayer, "Support vector machines for dyadic data," *Neural Computation*, vol. 18, no. 6, pp. 1472–1510, 2006.

[27] S. Hochreiter, T. Knebel, and K. Obermayer, "An SMO algorithm for the potential support vector machine," *Neural Computation*, vol. 20, no. 1, pp. 271–287, 2008.

[28] L. Solbach, R. Wöhrmann, and J. Kliewer, "The complex-valued continuous wavelet transform as a preprocessor for auditory scene analysis," in *Computational auditory scene analysis*, D. F. Rosenthal and H. G. Okuno, Eds. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1998, pp. 273–291.

[29] E. Levina and P. Bickel, "The earth mover's distance is the mallows distance: Some insights from statistics," in *IEEE International Conference on Computer Vision*, 2001.

[30] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43– 49, 1978.

Kamil Adiloğlu is a post doctoral research fellow at the METISS Group in INRIA, Rennes, France. He studied computer science and took his MS degree on algorithmic composition at the Middle East Technical University in Ankara, Turkiye. During his PhD studies, he worked on mathematical music theory. He received his PhD degree in 2009 from the Berlin Institute of Technology. He co-initiated the EU project "CLOSED" and worked on sparse audio features and sound classification during the project. Currently he develops statistical models on audio source separation and robust audio feature extraction.

Robert Anniés studied computer science at the Berlin Institute of Technology with specialization in machine learning and quantitative methods. He worked on audio classification in the EU project "CLOSED" and is now living and working in Switzerland. He is now with the ARTORG Center for Biomedical Engineering Research at the University of Bern as software architect.

Elio Wahlen was born in Germany on May 5, 1983. He received the Dipl.Ing. degree in media technology from the Hamburg University of Applied Sciences in 2008. In 2008 he was working at NIPS, TU-Berlin for the EU project "CLOSED". In 2009 his research interests moved towards intermedial theatre. He is currently finishing his M.A. in time based media at the Hamburg University of Applied Sciences.

Hendrik Purwins is lecturer at the Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain. He obtained his Ph.D. from Berlin Institute of Technology, with a scholarship from the "Studienstiftung des deutschen Volkes". Previously, he studied music at Berlin University of the Arts and mathematics at Bonn and Muenster University, achieving a diploma in pure mathematics. He has been visiting researcher at IRCAM, Paris, CCRMA, Stanford, and McGill University. He started playing the violin at age of 7. He has written more than 50 scientific papers and has won 12 research grants/prizes. His interests comprise statistical, unsupervised (online) models for machine listening, music generation, sound resynthesis, and failure prediction in semi-conductor manufacturing.

Klaus Obermayer received his Diplom degree in physics in 1987 from the University of Stuttgart, Germany, and the Dr. rer. nat. degree in 1992 from the Department of Physics, Technical University of Munich, Germany.

From 1992 and 1993 he was a postdoctoral fellow at the Rockefeller University, New York, and the Salk Institute for Biological Studies, La Jolla, USA. From 1994 to 1995 he was member of the Technische Fakultät, University of Bielefeld, Germany. He became associate professor in 1995 and full professor in 2001 at the Department of Electrical Engineering and Computer Science of the Berlin Institute of Technology, Germany. He is head of the Neural Information Processing Group and member of the steering committee of the Bernstein Center for Computational Neuroscience Berlin. He is also member of the governing board of the International Neural Network Society and was Vice-President of the Organisation for Computational Neuroscience from 2008-2011. From 1999-2003 he was one of the directors of the European Advanced Course of Computational Neuroscience. His current areas of research are computational neuroscience, artificial neural networks and machine learning with focus on pattern recognition applications, and the analysis of neural data. He co-authored more than 200 scientific publications.