Supervised Classification of Baboon Vocalizations

Maxime Janvier, Radu Horaud, Laurent Girin, Frédéric Berthommier,
Louis-Jean Boë, Caralyn Kemp, Arnaud Rey, Thierry Legou

Supervised Classification of Baboon Vocalizations

Maxime Janvier[*], **Radu Horaud**
INRIA Grenoble Rhône-Alpes
Grenoble, France
maxime.janvier@inria.fr
radu.horaud@inria.fr

Laurent Girin, Frédéric Berthommier, Louis-Jean Böe
GIPSA-lab
Grenoble-Alpes University, CNRS
Grenoble, France
laurent.girin@gipsa-lab.fr
frederic.berthommier@gipsa-lab.fr
louis-jean.boe@gipsa-lab.fr

Caralyn Kemp, Arnaud Rey
Laboratoire de Psychologie Cognitive
and Brain and Language Research Institute
Aix-Marseille University, CNRS
Marseille, France
caralyn@kemputer.com.au
arnaud.rey@univ-amu.fr

Thierry Legou
Laboratoire Parole et Langage
and Brain and Language Research Institute
Aix-Marseille University, CNRS
Marseille, France
thierry.legou@lpl-aix.fr

Abstract

This paper addresses automatic classification of baboon vocalizations. We considered six classes of sounds emitted by *Papio papio* baboons, and report the results of supervised classification carried out with different signal representations (audio features), classifiers, combinations and settings. Results show that up to 94.1% of correct recognition of pre-segmented elementary segments of vocalizations can be obtained using Mel-Frequency Cepstral Coefficients representation and Support Vector Machines classifiers. Results for other configurations are also presented and discussed, and a possible extension to the "Sound-spotting" problem, i.e. online joint detection and classification of a vocalization from a continuous audio stream is illustrated and discussed.

1 Introduction

Nonhuman primates produce a relatively limited variety of species-specific vocalizations in response to particular social events [1]. Until recently, classifying these vocalizations has been performed by ear and by time-consuming manual analysis [2]. Several researchers conducted various analyses, making comparison between studies, as well as between species, difficult [3]. The automatic classification of vocalizations can assist the field of primate communication in a multitude of ways: firstly, it can be used to assist and complement the classification made by experts. For example, it can be used to assess the relevance of different sets of acoustic features for the characterization of the different sound categories. Secondly, automatic classification of sounds in such a context can be exploited by audio/video recording systems dedicated to ethological studies or environment preservation. For example, the detection of relevant sounds emitted by the animals under study may indicate a scene of interest and trigger the video recording, thereby avoiding useless data storage and power consumption. Understanding the differences between the broad vocal classifications (i.e., comparison of a grunt to a scream) will better improve the fine-tuning of these analyses required for graded vocal calls and the differences in vocal production by different individuals for the same sound. In this work, we consider different supervised analyses for the classification of baboon vocalizations, which, to our knowledge, is the first study of its kind.

[*]M. Janvier is funded by the "Direction Générale de l'Armement" (DGA) included in the French Ministry of Defence.

In this paper we consider six categories of baboon vocalizations. We report the results obtained with the use of different audio signal representations and supervised classification methods to characterize and recognize these vocalizations. To this end, we tested different spectral features computed based on the usual short-term sliding window approach, e.g., Mel Frequency Cepstral Coefficients (MFCC). We propose to introduce a sparse subset of coefficients characterizing the harmonicity of the vocalizations, since, as opposed to (human) speech, the range of the fundamental frequency is quite different across the baboon sound categories. As for the classifiers, we used hidden Markov models (HMMs) [4] to model the dynamic evolution of the spectral patterns within each sound category. We also tested k-Nearest Neighbors (KNN) classifiers, Gaussian Mixture Models (GMM) and Support Vector Machines (SVM) [5], [6] with different configurations and appropriate preprocessing of the data (especially for time alignment of feature vector sequences). Note that most of the presented experiments concern isolated sounds that were manually pre-segmented, but we also discuss and illustrate the feasibility of the extension of our system(s) to the "soundspotting" problem, i.e. online joint automatic detection (i.e. segmentation) and classification of vocalizations from a continuous audio stream.

The paper is organized as follows: Section 2 describes the data that were used for this study; Sections 3 and 4 present respectively the different features and classifiers that were used; Experimental results are presented in Section 5 and conclusions are drawn in Section 6.

2 Data

We recorded the vocal behavior of *Papio papio* Guinea baboons housed at the Rousset-sur-Arc CNRS primate center, France. The vocalizations of sixteen baboons (13 females, 3 males; aged between 2 and 27 years at the start of recording) were considered for this study. Fourteen of the baboons were housed as part of a larger group in a 25 $\times$ 30 m outdoor enclosure connected by wire tunnels to indoor housing (6 $\times$ 4 m) used at night. The other baboons were housed separately in a 4.7 $\times$ 6.4m outdoor enclosures connected to indoor housing (2 $\times$ 4m). All groups had visual and auditory contact with each other. The monkeys could be identified by their individual physical characteristics and by number tags on a chain around their neck. Once daily feeding (fruits, vegetables and monkey chows) occurred at 5PM; water was provided ad libitum. See [7] for a more detailed description of the research facilities at the Rousset-sur-Arc CNRS primate center. We used opportunistic sampling techniques to record spontaneous vocalizations produced in response to social events and to stimuli occurring naturally within the baboons' environment. The presence of the recorders and their equipment did not disturb the baboons from their natural daily activities. Recording took place between 8:00 and 21:00 (except 17:00-18:00 due to the baboons being fed at this time) between September 2012 and June 2013. Recording was conducted at a distance from the baboons of $<$ 2m to 20m, with the greater distances suitable only for the long-distance vocalizations. A digital Zoom Handy Recorder H4n (Zoom, Japan: 44.1kHz sampling frequency, 16-bit resolution, mono) with a Me66 Sennheiser directional microphone (Sennheiser Electronic KG, Germany; with windscreen) was used to record the vocalizations. This is a super cardioid microphone with a high sensitivity (50 mV/Pa $\pm$ 2.5dB) and a wide (40Hz–20000Hz) and flat ($\pm$ 2.5dB) frequency response. As the vocalizations were recorded outdoors, environmental sounds at different noise levels may have interfered with the sounds at the focus of the recordings.

From continuous audio streams, individual "homogeneous" sequences of vocalizations (i.e. a series of sounds of the same class) were first manually extracted by an expert for analysis. Those sequences were further manually segmented into elementary sounds that were labelled to be submitted to our classifiers. Six vocalization types were considered in the present study: barks, grunts, copulation grunts (denoted "Cops" throughout the rest of the paper for concision), screams, wahoos, and yaks. In total, the number of sounds per classification was: 110 barks, 130 copulation grunts, 384 grunts, 119 screams, 64 wahoos, and 336 yaks. Original sequences were used to illustrate the feasibility of the "Sound-spotting" task (see Sections 4.4 and 5.4).

3 Features

This Section presents the audio features used in this study. Although we consider here vocalization *elements*, i.e. elementary sounds that can be part of a series of longer vocalizations, and that have been previously segmented, those elementary sounds can be of variable length. Moreover, they can

be more or less stationary (and in general, they are rather non stationary). Therefore, from these elementary sounds, we first extracted *time sequences* of *feature vectors* computed using a short-term sliding window (for instance, a 30ms-Hamming window with 50% overlap). This approach is familiar in speech processing, as well as in audio processing in general (e.g. for the analysis of domestic or environmental sounds), and we inspire from those fields. Also, the features that we use have been largely presented in the related literature [8, 9], and, thus, we present them only briefly.

Mel-Frequency Cepstral Coefficients: MFCCs [10] are cepstral coefficients that represent the envelope of the short-term spectrum on a perceptive mel-frequency scale. Those coefficients are computed as the discrete cosine transform (DCT) of the logarithm of FFT power coefficients passed through a mel-filter bank (e.g. 40 log-spaced bands in the range 300Hz-10kHz; the bandwidth and number of bands can vary; see Section 5). The first coefficient was omitted since it represents the absolute energy of the signal frame and not the spectral shape, and the 1st and 2nd derivatives are added optionnally (depending on experiment).

Average Spectral Features: We tested a series of features that represent average properties of the Short-Term Fourier Transform (STFT) spectrum. The *Spectral Roll-off* is the cut-off frequency below which 99% the spectral energy is contained. The *Spectral Moments* characterize the overall shape of the spectrum using n-order moments of frequency bin weighted by spectral magnitude. We tested the 4 first moments. The *Spectral Slope / Decrease* represents the global amount of decreasing of the spectral amplitude. The *Spectral Flatness* of the magnitude spectrum is given by the ratio between its arithmetic and geometric mean. Finally, the *Spectral Flux / Correlation* measure the average variation between two consecutive spectra.

F_0 and Harmonicity Index: The above-mentioned MFCCs (resp. the ASF) are coefficients that characterize the spectral envelope (resp. the global shape of the spectrum) on a perceptive (resp. linear) frequency scale. MFCCs are widely used in Automatic Speech Recognition (ASR) systems [10] since the spectral envelope characterizes the different speech sounds through the effect of the speaker's vocal tract, while cutting loose of speech sound dependence on fundamental frequency F_0. This is a desirable property for ASR, in order to limit speech variablity across speakers and utterances. In contrast, in the present context of baboon vocalizations, we think that the F_0 *range* can be a discriminative feature since it varies much between some of the considered classes. Therefore, we propose to test the F_0 value (also extracted on a short-term basis) as an audio feature. We also tested the *harmonicity index*, which is the ratio between the second maximum of the signal (short-term) autocorrelation function (which is also used to detect F_0) and the maximum which is obtained at lag zero. The harmonicity index provides some simple confidence measure of the F_0 value.

Feature post-processing: The successive feature vectors of a sound can be further processed to produce different final features, which will feed the classifiers. In particular, the feature vector sequences are generally of different lengths, whereas some of the tested classifiers (KNN, GMMs and SVMs; see Section 4) are designed to process fixed-size vectors (or fixed-size sequences of vectors reorganized as vectors). Therefore, it is necessary adress the problem of time normalization. In the present study, we consider two simple forms of time normalization. The first one consists of *averaging* the vectors in the time dimension over the entire acoustic event. Therefore, the feature vector sequence is replaced with a single mean feature vector (the standard deviation can also be used). The second form regards the *interpolation* of the feature vector sequence to the class' average duration, using basic (e.g. spline) interpolation/resampling techniques. Note that the GMM-T and HMMs classifiers are fed directly with the original feature vector sequence and do not need time normalization (HMMs are specifically designed to model dynamic sound representations). Note finally that the final representation may consist of the (row-wise) concatenation of different features. This is a particular case of information fusion for classification (see Section 4.3).

Implementation The MFCC and ASF features have been computed with the Python/C++ toolbox YAAFE [9]. The F_0 and harmonicity index analysis function was conducted using our own Matlab implementation.

4 Classifiers

4.1 Definition

A multiclass classifier consists of a mapping $g : \mathcal{X} \times \mathcal{C} \to \mathbb{R}$, whereby $\mathcal{X}$ is the feature space, $\mathcal{C} = \{1, \ldots, C\}$ is the set of labels and C is the number of classes. The dimension of $\mathcal{X}$ may be

fixed or varying with the sound, depending on the feature used. Given a feature vector (or sequence of feature vectors) $\boldsymbol{x} \in \mathcal{X}$, $g(\boldsymbol{x}; c)$ is the score of classifying $\boldsymbol{x}$ as c. A new unlabeled observation $\boldsymbol{x} \in \mathcal{X}$ is classified as: $c^*(\boldsymbol{x}) = \arg\max_{c \in \mathcal{C}} g(\boldsymbol{x}; c)$. $\mathbf{X}$ will denote the training set, i.e. a set of feature vectors $\mathbf{X} = \{\boldsymbol{x}_n\}_{n=1}^N$ whose class is known, used to train the classifiers.

4.2 Four Classifiers

In this section, we present the four types of classifiers that were used in the present study. As some features are commonly used in speech and audio processing and the Signal Processing / Machine Learning communities, we present them very briefly, with links to the related literature.

k-nearest neighbors (KNN): The KNN classifier first find the subset $S_k(\boldsymbol{x}) \subset \mathbf{X}$ containing the k closest points to a given vector $\boldsymbol{x}$. $g_{k\text{NN}}(\boldsymbol{x}, c)$ is then the number of feature vectors among $S_k(\boldsymbol{x})$ that belong to the class c.

Support Vector Machines (SVMs): SVMs are a discriminative binary classification method (see [5] for a detailed description), which has already been used in sound recognition, e.g. [6, 11]. SVMs provide a discriminative function $h(\boldsymbol{x})$, learnt form a set of positive examples and a set of negative examples. The points satisfying $h(\boldsymbol{x}) = 0$ form a hyperplane in the space induced by a chosen kernel function $k(\cdot, \cdot)$. $h(\boldsymbol{x}) > 0$ means that $\boldsymbol{x}$ should be classified as positive and $h(\boldsymbol{x}) < 0$ as negative. The multi-class task uses *one-versus-rest* strategy. Also, we tested four different kernels (linear, radial basis, polynomial and sigmoid).

Gaussian Mixture Models (GMM): A GMM is a probabilistic generative model widely used in classification tasks [5]. Here, we use one GMM per sound class, which is a weighted sum of M Gaussian components. The parameter set λ_c is composed of M weights, mean vectors and covariance matrices. We thus train C sets of parameters using the well-known Expectation-Maximization (EM) algorithm. The mapping g corresponds to the likelihood of the observed data given the model parameters. GMMs can be applied directly on the mean feature vector (in such case, we simply denote this configuration with GMMs). Alternately, for a sequence of feature vectors $\boldsymbol{x} = [\boldsymbol{x}^1, \ldots, \boldsymbol{x}^T]$, which are assumed to be independent, we calculate: $g_{\text{GMM}}(\boldsymbol{x}; c) = \text{p}(\boldsymbol{x}|\lambda_c) = \prod_{t=1}^T \text{p}(\boldsymbol{x}^t|\lambda_c)$. We denote this configuration by GMMs-T.

Hidden Markov Models (HMM): HMMs also belong to the family of generative models [5, 10]. In an HMM, the observations depend on a hidden discrete random variable called state, taking S values. The state sequence is assumed to be a first-order "left-to-right" Markovian process and the emission probability is a GMM. Thus, the model consists of the parameters of the GMMs and the parameters modeling the Markovian dynamics. All are learnt using the EM algorithm. The function g is also the likelihood of the observations given the model: $g_{\text{HMM}}(\boldsymbol{x}; c) = \text{p}(\boldsymbol{x}|\xi_c)$.

Implementations: We used the standard Matlab KNN and GMMs algorithms. The HMMs are from the PMTK3 library [12]. The SVMs are implemented using libSVM [13].

4.3 Information Fusion

In Section 3, we have seen that several kinds of features can be extracted from the baboon vocalizationsl to describe their spectro-temporal characteristics in order to be used in a supervised classification scheme. This naturally raises the question of combining those features into a multi-modal/multichannel classifier that would optimally exploit all information in an efficient manner, a problem sometimes referred to as *sensor fusion*. This makes particular sense in the present study, since we postulated in Section 3 that, as opposed to ASR, the F_0 information is expected to provide significant information about sound class, it is therefore necessary to test if this information can be used in a complementary way to the spectral envelope (for instance MFCCs) information.

The usual, and simplest approach, known as *early integration*, consists in the (row-wise) concatenation of the different features (or feature vectors) into a single vector (in which dimension is equal to the sum of the dimensions of the original feature vectors), possibly integrating some cross-modal normalization processes. This new representation can then be used directly with the different classifiers presented above. In contrast, *late integration* performs the fusion of the features at the decision level of separate classifiers [14]. Thus, a different classifier (of same or different type) can be used on each feature vector and then the outputs (crisp decision, confidence score, log-likelihood values etc.) of these classifiers are merged using a higher level process. Finally, we can consider an

intermediary common space for fusion which is neither the input space nor the output space, leading to a type of *mid-level integration*. In particular, in the field of kernel-based classifiers (such as SVMs), a new state-of-the-art fusion strategy has emerged called Multiple Kernel Learning [15]. In this approach, the fusion is made "inside" the classifier: the kernel of the classifier is computed as a combination of multiple kernels, for instance, one kernel for each feature. One advantage is the ability to choose one type of kernel and its parameters according to the features. In Section 5.3, we will test this strategy for the integration of MFCCs and F_0 features in the present task of baboon vocalization classification.

4.4 The "Sound-spotting" Task

The above techniques are all applied on elementary sounds manually extracted from vocalization sequences. In practice, it is desirable to have a system that is able to automatically perform both detection (i.e. segmentation of a series of vocalizations into elementary sounds) and classification of the detected elementary sounds from the continuous audio stream. This task can be referred to as "Sound-spotting", in reference to the "Word-spotting" task in ASR which is the detection of keywords in continuous speech signals. A naive but efficient strategy consists of applying any of the previous classifiers (that have been tuned on a training corpus of elementary sounds) on a sliding window and decide of the detection if some criterion (e.g. a likelihood function), provided by the classifier, exceeds a given threshold. Temporal integration is necessary to make this joint detection/classification robust, and this can be done at the criterion level (e.g. by averaging frame-wise likelihoods) or at the feature level (e.g. by varying the sliding window length)[1]. In the present paper, we did not conduct a deeper investigation of the "Sound-spotting" problem, but in Section 5.3, we present some elements which illustrate the feasibility of this task using the proposed classifiers.

5 Experiments

5.1 Setup

Given the database described in Section 2, different combinations of features, post-processing and classifiers have been tested. We performed 5-cross validation tests, and used the accuracy score as a metric of the performance in order to be able to statistically compare the different configurations. For each experiment reported in the next section, only the best configuration of parameters (using grid search and cross validation) has been retained due to the large number of parameters involved. For the features, MFCCs reached its best results using 20 coefficients (with the first one omited), with the derivates at the first and second order on a 10Hz-10000Hz bandwidth. As for the classifiers, SVMs have shown the best results using linear kernels and radial basis kernels with a regulation parameter equal to 0.1 and the one-versus-rest strategy. HMM have been tested with 3 to 8 states and 5 to 10 components per state. Best results with GMM-based methods needed between 5 and 10 components in the mixture.

5.2 Results with Individual Feature Sets

We first present the results obtained separately with the different feature sets, i.e. either MFCCs or ASF or F_0+harmonicity index. The accuracy scores are given in Table 1 for a selected set of configurations, and confusion matrices are given in Table 2 and Table 3 for a subset of those configurations.

The best performance are obtained with SVMs (with a radial basis kernel) applied on averaged MFCC coefficients, with an accuracy score of 94.1%. This is a very good result, even for the limited number of classes of the present problem, since there is no a priori reason to think that a vast majority of the elementary sounds of the six classes are clearly prone to discrimination: This is actually a major outcome of the present study. The confusion matrix for this configuration (Table 2b is well balanced, with no major class confusion. Best results per class are obtained for Barks (97.3% accuracy) and worst result per class are obtained for Cops with 83.8% accuracy, and 13.8% of confusion with Grunts. It is important to note that SVMs are here applied on an averaged MFCC

[1]This is reminiscent of the "early" vs "late" integration problem discussed in Section 4.3, but considering here temporal fusion and not feature fusion.

vector (to represent the whole sound). Hence, the time structure of the spectral vector sequence does not seem to be very important, leastways not as important as in speech (even if we compare with a short word recognition task). This is confirmed by the score of the SVMs applied on time-interpolated MFCC vectors, which is a bit lower than with averaged MFCC vectors at 90.5%. And this is more severely confirmed by the scores obtained with the HMMs applied on the original MFCC vector sequences (see Table 2a): the accuracy score is here only 80.8%, which is quite deceiving. The confusion matrix exhibits notable confusions from Cops to Barks and to Grunts, and from Grunts to Cops (but not from Barks to Cops), and also from Yaks to Screams, which is surprising. This not only suggests that there is relatively poor additional information in the vector *sequence* compared to the vector mean for the task at hand, but it also suggests that the HMMs are not an appropriate tool for the modeling of such type of sounds. The latter makes sense since it is not clear so far if there exists a phonological structure in the baboon vocalizations that could be efficiently exploited by the state-space modeling of HMMs[2]. Finally, GMMs (92.7% accuracy; Table 2d) and KNN (92.4% accuracy; Table 2c), both applied on averaged vectors, are a bit below SVMs, confirming that most of the discriminative information is contained in the average vector, and that good recognition scores can be obtained with relatively basic classifiers. KNN applied on interpolated MFCC vectors are at 93.1% accuracy[3], and we did not test GMMs on interpolated MFCCs to avoid the "curse of dimensionality" problem which is typical for this model.

The scores obtained with ASF features are very deceiving. Many different combinations of ASF features were tested (with the different classifiers), and the best accuracy score is 73.2% obtained with SVMs on average ASF vectors (hence we only report this configuration in Table 1). Moreover, when using concatenation of MFCCs and ASF features (i.e. basic "early" fusion at the feature level, see Section 4.3), the scores do not improve significantly compared to using only the MFCCs, they even decrease in some configurations (that is the case for SVMs, see Table 1). Therefore, the ASF do not complement the MFCC information, which was predictable (they provide information on the global shape of the spectrum with generally less resolution than MFCCs, provided that the cepstral model order is sufficiently large). Therefore, we did not further consider those ASF features.

Generally, the results obtained with F_0 alone or F_0 concatenated with the harmonicity index are remarkable, given that it is quite rudimentary information. Here, the best results are obtained with the SVMs applied on interpolated F_0 vectors, which reach 71.0%. GMM-T comes a very close second with 70.9% accuracy. Both exploit temporal information (from interpolated or original vector sequence), but the accuracy score of the SVMs applied on the average F_0 vector is also very close at 69.6% accuracy. However, the confusion matrices for the two latter two configurations differ significantly: the matrix for GMM-T (Table 3a) is more balanced, whereas the matrix for SVMs (Table 3b) shows that the Grunts and Yaks have better results, while the Wahoos are totally confused (mainly with Barks and Cops) which is surprising. This can be explained partly by the fact that Wahoos have some prosody which is reduced by the averaging process. Note that the SVMs scores are biased by the fact that the best classification is obtained for the two classes with the higher cardinals (Grunts and Yaks), and only 3 classes out of 6 can actually be regarded as "correctly" classified. In contrast, the more well-balanced GMM-T matrix exhibits 5 classes out of 6 being fairly well classified. GMMs (68.1% accuracy; confusion matrix in Table 3d) and KNN (65.4% accuracy; confusion matrix in Table 3c), both applied on average F_0 features, are a bit below the others classifiers using F_0 as a feature, but not much. KNN applied on interpolated F_0 vectors are at 69.8% accuracy. Therefore, here also, the different classifiers for "fixed-size" features in both average and interpolated configurations are quite close to each other. Altogether, those results show that basic information about harmonicity (say F_0 range + harmonicity confidence) is enough to provide honorable classification of 6-class baboon vocalizations. Note that HMMS are, again, deceiving, with only 45.3% of correct classification.

5.3 Results with Kernel-Based Fusion of MFCCs and F_0

As announced in Section 4.3, we report the results obtained with the *mid-level integration* of MFCC and F_0 features, using fusion of SVMs kernels. As an example, Table 4 shows the results of a Multi-

[2]However, the GMMT score is also deceiving (78.5% accuracy) hence possibly pointing a problem with the use of the original MFCC sequence, and so far we cannot clearly explain this result.

[3]Hence, KNN with interpolated MFCCs is a bit better than KNN with averaged MFCCs, whereas SVMs with interpolated MFCCs is a bit lower than SVMs with averaged MFCCs. Altogether, the scores with KNN, SVMs and GMMs applied on either averaged or interpolated MFCCs are quite close to each other.

Features	Classifier	Representation	Accuracy
MFCCs	KNN	Averaging	$92.4\% \pm 2.9\%$
MFCCs	SVMs	Averaging	$94.1\% \pm 1.2\%$
MFCCs	GMMs	Averaging	$92.7\% \pm 1.8\%$
MFCCs	KNN	Interpolation	$93.1\% \pm 3.0\%$
MFCCs	SVMs	Interpolation	$90.5\% \pm 2.9\%$
MFCCs	GMMs-T	Sequencing	$78.5\% \pm 4.8\%$
MFCCs	HMMs	Sequencing	$80.8\% \pm 3.9\%$
ASF	SVMs	Averaging	$73.2\% \pm 2.3\%$
MFCCs & ASF	SVMs	Averaging	$92.4\% \pm 2.7\%$
F_0	KNN	Averaging	$65.4\% \pm 6.9\%$
F_0	SVMs	Averaging	$69.6\% \pm 2.7\%$
F_0	GMMs	Averaging	$68.1\% \pm 7.4\%$
F_0	KNN	Interpolation	$69.8\% \pm 4.6\%$
F_0	SVMs	Interpolation	$71.0\% \pm 2.3\%$
F_0	GMMs-T	Sequencing	$70.9\% \pm 4.2\%$
F_0	HMMs	Sequencing	$45.3\% \pm 7.3\%$

Table 1: Accuracy score for different combinations of audio features, post-processing, and classifiers. "Sequencing" refers to using the original sequence of vectors.

	barks	cops	grunts	screams	wahoos	yaks
barks	**102**	0	0	1	7	0
cops	13	**89**	12	7	7	2
grunts	3	42	**312**	12	11	4
screams	1	0	0	**114**	0	4
wahoos	9	0	0	0	**55**	0
yaks	7	9	8	48	12	**252**

(a) Hidden Markov Models (HMMs)

	barks	cops	grunts	screams	wahoos	yaks
barks	**107**	0	1	0	1	1
cops	0	**109**	18	1	0	2
grunts	1	6	**369**	0	1	7
screams	0	0	0	**114**	1	4
wahoos	6	0	0	0	**58**	0
yaks	0	5	11	1	0	**319**

(b) Support Vector Machines (SVMs)

	barks	cops	grunts	screams	wahoos	yaks
barks	**106**	0	0	0	2	2
cops	3	**104**	17	1	3	2
grunts	1	9	**367**	0	0	7
screams	0	0	0	**109**	0	10
wahoos	6	0	0	0	**58**	0
yaks	0	1	2	20	1	**312**

(c) k-Nearest Neighbors (KNN)

	barks	cops	grunts	screams	wahoos	yaks
barks	**105**	0	0	3	0	2
cops	0	**115**	8	1	0	6
grunts	0	19	**350**	2	0	13
screams	0	0	1	**112**	0	6
wahoos	5	0	0	1	**57**	1
yaks	1	2	9	3	0	**321**

(d) Gaussian Mixture Models (GMMs)

Table 2: Confusion matrix for the baboon vocalization recognition systems using average Mel-frequency cepstral coefficients (MFCCs) as features for SVMs, GMMs and KNN, and using original sequence of MFCCs for HMMs.

	barks	cops	grunts	screams	wahoos	yaks
barks	**85**	0	1	0	18	6
cops	17	**29**	44	1	36	3
grunts	9	26	**326**	0	17	6
screams	1	0	0	**90**	1	27
wahoos	14	7	0	0	**43**	0
yaks	37	7	7	32	16	**237**

(a) Gaussian Mixture Models on a sequence of vectors (GMM-T)

	barks	cops	grunts	screams	wahoos	yaks
barks	**83**	12	1	0	0	14
cops	32	**19**	72	0	0	7
grunts	13	1	**363**	0	0	7
screams	2	0	0	**67**	0	50
wahoos	29	24	8	0	**0**	3
yaks	43	3	9	18	0	**263**

(b) Support Vector Machines (SVMs)

	barks	cops	grunts	screams	wahoos	yaks
barks	**65**	13	3	0	7	22
cops	30	**30**	53	0	10	7
grunts	9	36	**328**	0	3	8
screams	1	0	1	**69**	0	48
wahoos	28	9	4	0	**14**	9
yaks	34	11	12	30	7	**242**

(c) k-Nearest Neighbors (KNN)

	barks	cops	grunts	screams	wahoos	yaks
barks	**74**	0	1	0	29	6
cops	21	**18**	50	0	35	6
grunts	6	20	**338**	0	12	8
screams	2	0	0	**94**	1	22
wahoos	9	3	3	0	**48**	1
yaks	45	1	8	56	20	**206**

(d) Gaussian Mixture Models (GMMs)

Table 3: Confusion matrix for the baboon vocalization recognition systems using average F_0 (fundamental frequency) as feature for SVMs, GMMs and KNN, and using original sequence of F_0 for GMM-T.

ple Kernel Learning experiment, in which a linear kernel has been trained on MFCC features, while another linear kernel has been trained on F_0 features, and the combination of those kernels has been computed and used in a third SVM. It can be seen that this configuration does not outperform the SVMs which uses only MFCCs as features: the accuracy scores are $88.1\% \pm 2.9\%$ for the former vs $91.2\% \pm 3.3\%$ for the latter[4]. None of the other tested configurations of kernels and hyper parameters have shown a significant improvement. One conclusion of this experiment is that, although the F_0 (and harmonicity index) feature separately carries a significant information which is exploitable for the automatic recognition of baboon vocalization, this feature was not shown in our experiments to be complementary to the MFCC features for this task. On the contrary, the combination of F_0 and MFCCs only lead so far to slightly decrease the scores obtained with MFCCs alone, which is a bit deceiving. Of course, this is also because MFCC representation initially led to impressive scores. Further investigation of the characterization of those features for the baboons vocalizations is necessary to precisely describe the redundancy between them and confirm the seeming absence of complementarity which has been observed in our experiments.

5.4 Feasibility of Sound-Spotting

In this subsection, we illustrate the feasibility of the Sound-spotting task described in Section 4.4 by applying the SVMs of Section 4.2 on an example of original (i.e. unsegmented) sequence. The SVMs were fed with MFCC vectors on a frame-by-frame basis (i.e. average of one vector at a time, corresponding to a 200ms-frame of signal, with 10ms-hop size). For each frame and class c, we retrieved $p(c|\boldsymbol{x})$ the posterior probability of the frame being part of a vocalization of class c given the input MFCC vector $\boldsymbol{x}$, which is the criterion used by the SVMs for classification [16]. Fig. 1 shows the results of this analysis. The top subfigure shows an excerpt of a vocalization waveform with the corresponding class boundaries and labels which were manually annotated. The three other subfigures plot the values of $p(c|\boldsymbol{x})$ for the Barks, Grunts, Screams and Yaks, respectively (from top to bottom; probabilities for Cops and Wahoos are not displayed for clarity). It is evident that the probability contours quite well with the actual classes, i.e. globally, the probability values are high when the corresponding class is emitted, and low when another class or background noise is

[4]This latter score is different (a bit lower) than the SVMs/MFCCs score of Table 1 because a radial basis kernel was used in the SVMs of Section 5.2.

	barks			cops			grunts			screams			wahous			yaks		
barks	**60**	**107**	**60**	13	0	2	6	1	0	0	0	0	0	0	2	31	2	2
cops	26	1	7	**13**	**92**	**82**	81	31	36	0	1	1	0	0	1	10	5	3
grunts	11	1	2	3	7	4	**363**	**369**	**371**	0	0	1	0	1	0	7	6	6
screams	1	1	0	1	0	1	0	0	0	**94**	**109**	**96**	0	0	0	54	9	22
wahous	23	9	7	20	0	0	17	0	0	0	0	0	**0**	**55**	**55**	4	0	2
yaks	32	1	0	4	3	5	10	14	14	18	8	14	0	0	4	**272**	**310**	**299**

Table 4: Confusion matrix for one instance of Multiple-Kernel SVMs combining MFCC and F_0 features. For each cell, the three numbers from the left to the right corresponds to the result of classification for: (1) SVMs with a linear kernel on F_0, (2) SVMs with a linear kernel on MFCCs, (3) SVMs with a combination of the two precedent kernels.

emitted. For this example, a very simple detection strategy based on thresholding can be applied: Class c is detected as $p(c|\boldsymbol{x}) > 0.5$ (the probabilities for the different classes sum up to 1, hence only one class at a time can be detected). Merging the successive frames associated with the same class leads to the detected boundaries represented in the top subfigure of Fig. 4.4 with background color corresponding to the probability contours. The detection is fairly good but not perfect: for example, background noise is confused with Grunts at approx. 6s, and the boundaries between Yaks and Screams are not easy to define (nor is it easy for the human listener in this example, and manual labeling may actually be inacurate). Moreover, many sequences are not so clear. However, more refined strategies for time integration of frame-wise information, such as the ones mentioned in Section 4.4, are expected to fix these problems and be more robust in general. Part of our future work is to explore such strategies and derive an efficient and robust Sound-spotting algorithm in the present problem of baboon vocalization recognition.

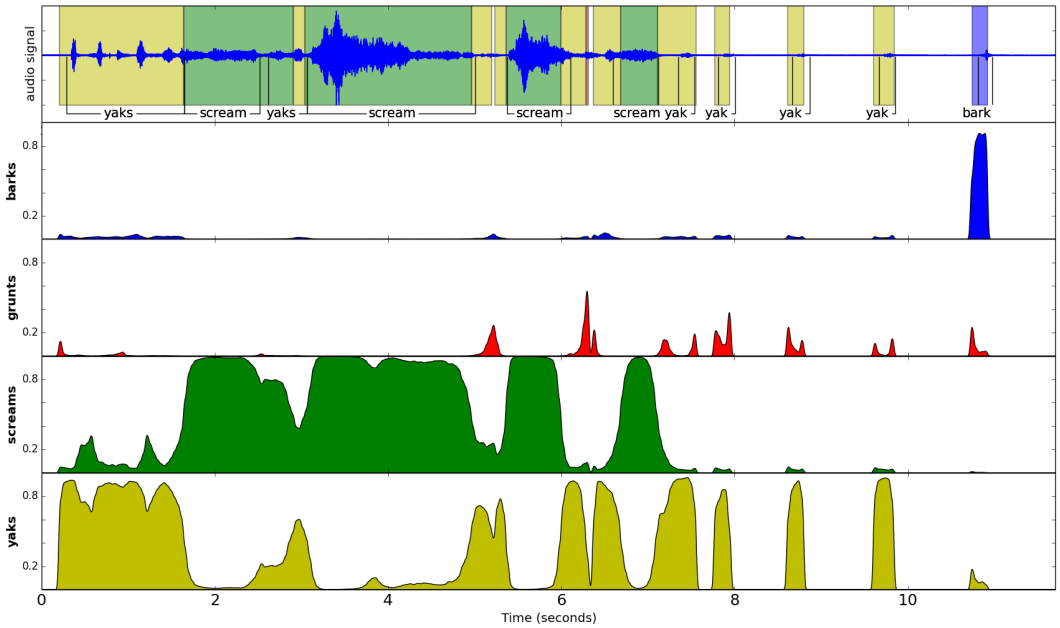

Figure 1: Example of automatic joint segmentation and classification using the SVMs of Section 4.2 (see text for details).

6 Conclusion

In this paper we have adressed the problem of automatic classification of Guinea baboon vocalizations. Six classes of sounds have been considered, and experiments have shown that several types of classifier (KNN, GMM, SVM) lead to correct classification scores higher than 90% for pre-segmented elementary vocalizations. The higher scores were obtained with SVMs applied on

average MFCC vectors (94.1% accuracy), and the principal remaining confusions were observed to be between grunts and copulations grunts. It is not entirely surprising that the classifiers have difficulty in distinguishing these two vocalizations; of all the sound classes, the call units of these two are the most similar from both an auditory perception and acoustic structure standpoint. This study has also shown that the fundamental frequency F_0 (alone or coupled with harmonicity index) has a significant discriminative power: several classifiers applied on these features provided approximately 70% correct classification. Indeed, analysis of the baboon vocal repertoire shows that the baboons strongly modulate their F0 between vocalizations, particularly between short- and long-distance vocal categories (Kemp et al., in prep.). However, and quite deceivingly, this information was not found to be complementary to the spectral envelope information in our study. Finally, although we did not conduct a deep investigation of the Sound-spotting problem in the present study, the observation of the good behavior of classifiers, designed on elementary sounds when applied on continuous audio streams, shows that joint segmentation and recognition is expected to be feasible with a well-grounded time integration process. This time integration can be processed at the feature level, at the classifier output level, or at some "mid-level" within the classifier, echoing the discussion of Section 4.3 on feature information fusion. Future work will concern this task, which is essential to design a real-world system. We will also consider increasing the number of classes and defining confidence measures to help the exploitation of the classification results in primatology studies.

Acknowledgments: Yannick Becker and the staff of the Rousset-sur-Arc primate center are acknowledged for technical support.

References

[1] K. Hammerschmidt and J. Fischer, "Constraints in primate vocal production," *The evolution of communicative creativity: From fixed signals to contextual flexibility*, pp. 93–119, 2008.

[2] A. Mielke and K. Zuberbühler, "A method for automated individual, species and call type recognition in free-ranging animals," *Animal Behaviour*, vol. 86, no. 2, pp. 475–482, 2013.

[3] P. Maciej, J. Fischer, and K. Hammerschmidt, "Transmission characteristics of primate vocalizations: implications for acoustic analyses," *PloS one*, vol. 6, no. 8, p. e23015, 2011.

[4] L. Deng and X. Li, "Machine learning paradigms for speech recognition: An overview," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 5, pp. 1060–1089, 2013.

[5] C. M. Bishop, *Pattern recognition and Machine learning*. Springer New York, 2006.

[6] G. Guo and S. Z. Li, "Content-based audio classification and retrieval by support vector machines," *IEEE Trans. on Neural Networks*, vol. 14, no. 1, pp. 209–215, 2003.

[7] J. Fagot and E. Bonté, "Automated testing of cognitive performance in monkeys: Use of a battery of computerized test systems by a troop of semi-free-ranging baboons (papio papio)," *Behavior Research Methods*, vol. 42, no. 2, pp. 507–516, 2010.

[8] G. Peeters, "A large set of audio features for sound description (similarity and classification) in the cuidado project," 2004.

[9] B. Mathieu, S. Essid, T. Fillon, J. Prado, and G. Richard, "Yaafe, an easy to use and efficient audio feature extraction software," in *Int. Conf. for Music Information Retrieval (ISMIR)*, 2010.

[10] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[11] A. Temko and C. Nadeu, "Classification of acoustic events using SVM-based clustering schemes," *Pattern Recognition*, vol. 39, no. 4, pp. 682–694, 2006.

[12] K. P. Murphy, *Machine learning: A probabilistic perspective*. MIT Press Boston, 2012.

[13] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[14] L. I. Kuncheva, J. C. Bezdek, and R. P. Duin, "Decision templates for multiple classifier fusion: an experimental comparison," *Pattern Recognition*, vol. 34, no. 2, pp. 299–314, 2001.

[15] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *The Journal of Machine Learning Research*, pp. 2211–2268, 2011.

[16] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *The Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.