Sparse Representation-Based Classification of Mysticete Calls

Thomas Guilment, François-Xavier Socheleau, Dominique Pastor, Simon Vallez

Sparse Representation-Based Classification of Mysticete Calls

Thomas Guilment,* Francois-Xavier Socheleau,† and Dominique Pastor‡

IMT Atlantique, Lab-STICC, Bretagne Loire University,

Technopole Brest-Iroise CS83818, Brest 29238, France

Simon Vallez§

Sercel, 12 Rue de la Villeneuve, 29200 Brest, France

(Dated: October 9, 2018)

Abstract

This paper presents an automatic classification method dedicated to mysticete calls. This method relies on sparse representations which assume that mysticete calls lie in a linear subspace described by a dictionary-based representation. The classifier accounts for noise by refusing to assign the observed signal to a given class if it is not included into the linear subspace spanned by the dictionaries of mysticete calls. Rejection of noise is achieved without feature learning. In addition, the proposed method is modular in that, call classes can be appended to or removed from the classifier without requiring retraining. The classifier is easy to design since it relies on a few parameters. Experiments on five types of mysticete calls are presented. It includes Antarctic blue whale Z-calls, two types of "Madagascar" pygmy blue whale calls, fin whale 20 Hz calls and North-Pacific blue whale D-calls. On this dataset, containing 2185 calls and 15000 noise samples, an average recall of 96.4% is obtained and 93.3% of the noise data (persistent and transient) are correctly rejected by the classifier.

PACS numbers: PACS: 30.Sf, 30.Wi

* thomas.guilment@imt-atlantique.fr; Corresponding author.

† fx.socheleau@imt-atlantique.fr

‡ dominique.pastor@imt-atlantique.fr

§ simon.vallez@sercel.com

I. INTRODUCTION

Passive acoustic monitoring (PAM) is very useful tool for helping scientists study marine mammals [1], detect their presence during seismic surveys and as a consequence, mitigate the impact of man-made acoustic activities [2, 3]. The success of PAM has led to an increasing deployment of underwater acoustic recorders across many oceans [4]. As a result, the development of efficient and robust automatic methods is needed to analyze the growing amount of acoustic data generated by these recording systems. Such methods are helpful for human analysts to detect, classify, locate, track or count marine mammals.

PAM is particularly relevant for mysticetes or baleen whales which are known to produce a wide variety of underwater sounds [5–7]. Their repertoire is composed of tonal [8, 9], frequency-modulated (FM) [10], pulsive [11, 12] sounds and other calls with exotic names such as boings [13], moans and grunts [14], exhalation and gunshot [15], and "star-wars" vocalization [16]. Mysticete calls exhibit different levels of variability. Some calls, such as Antarctic blue whale Z-calls [17], only show slight inter-annual and seasonal variations [8], whereas other vocalizations, such as songs produced by bowhead whales [3, 18], fully change from one year to another [19]. In between, there are a variety of calls with the same signal structure but with parameters, such as duration and/or bandwidth and/or FM rate, whose values may change over time [7].

Automatic classifiers of mysticete calls face several challenges. As any pattern recognition algorithms, they have to identify the salient features of the calls of interest. However, this may be difficult because (i) signal-to-noise ratios can be low, (ii) propagation effects can distort the call features [20] and, (iii) the selected features must not only describe and discriminate the calls of interest, but also [21] *"provide contrast to any other type of signal that is likely to occur"* in the same acoustic context. Past experiments have shown that acoustic recordings can contain a wide variety of interfering transient sounds in the frequency range of mysticete calls [22–26]. Therefore, providing classifiers with a rejection option that refuses to assign a signal of no interest to any class is of prime importance for PAM applications.

In the context of multiclass classification, most automated techniques for mysticete calls implement a two-step procedure. They usually operate in the frequency or cepstral domain and first extract sound attributes like start frequency, end frequency, frequency slope, duration etc. A supervised learning algorithm then maps these attributes to a call class after learning training examples labeled by human analysts. Classifier of this kind include aural classification [27], neural networks

[3], hidden Markov models [28], quadratic discriminant function analysis [29], Gaussian mixture models [30] or classification trees [31]. More recently, Halkias *et al.* [25] proposed an alternative approach based on hybrid generative/discriminative models commonly used in machine learning. This method involves injecting a spectrogram image of the sound to process into a multiple-layer neural network. The main advantage of the used network is that it automatically learns the signal attributes from unlabeled data and does not rely on "hand-engineered" features.

Although applied with success in specific contexts, state-of-the-art methods may however show some limitations. For instance, some classifiers lack of general applicability because they are tuned for specific species. This is the case of spectrogram correlation [32], non-spectrogram correlation [13], vector quantization algorithm and dynamic time warping [33]. Others may require to tune many (hyper)-parameters [25, 29]. In case these parameters are not easy to physically interpret, their numerical values may be difficult to set, which can limit the robustness of the classifier or lead to under- or over-fitting. Moreover, some methods offer a rejection option that rely on parametric models of noise [24] or require the classifier to learn the features of the unwanted signals [25]. Exhaustive noise learning or modeling is hardly feasible in practice since the underwater acoustic environment is very complex and contains many transient signals with very different features. In addition, these features may fluctuate in time and space so that they may greatly vary from one dataset to another. Finally, most existing classifiers lack of modularity/flexibility and are often designed for a specific set of calls, so that adding or removing a call class usually requires to "retrain" the entire classifier. In a PAM context, where the same classifier may be used on platforms operating at different geographic locations and at different time of the year, offering the capability of selecting online the class of calls taken into account by the classifier may have an operational interest. Classes corresponding to species whose habitats are known to be far away from the sensor may therefore be removed from the classifier, thus reducing the probability of miss-classification.

In this paper, a general method capable of classifying multiple mysticete calls is described. The method has been designed to meet the following requirements: (i) a rejection option is implemented, (ii) the classifier is modular, (iii) it is tuned by a very few (easy-to-set) parameters and (iv) it involves a compression option so as to provide a good trade-off between robustness to call variability and computational load. The proposed approach relies on the sparse framework recently developed in signal processing and machine learning [34–36]. Sparse representations express a given signal as a linear combination of base elements in which many of the coefficients are zero. Such representations can capture the possible variability observed for some vocalizations and can automatically be learned from the time-series of the digitized acoustic signals, without requiring prior transforms such as spectrograms, wavelets or cepstrums. This framework is general and applicable to any mysticete call lying in a linear subspace described by a dictionary-based representation. Successfully applied to the detection of mysticete calls [23], this framework is thus extended to the classification of mysticete calls and evaluated in this context. To the authors' best knowledge, this paper is a first attempt in this direction.

The paper is organized as follows. In Sec. II, the classification method is presented. The performance of the classifier is then evaluated on five call classes extracted from four real datasets in Sec. III. Finally, conclusions are given in Sec. IV.

Notation: Throughout this paper, $\mathbb{R}^n$ designates the space of all n-dimensional real column vectors and $\mathbb{R}^{n\times m}$ is the set of all real matrices with n rows and m columns. The superscript ^{T} means transposition. $\|\cdot\|_p$ designates the ℓ_p norm.

II. METHODOLOGY

Supervised learning makes it possible for systems to perform automatic classification of previously unseen inputs, after learning examples labeled by experts. The learning phase proceeds as follows. A labeled or training dataset is made of N pairs $\{(\boldsymbol{s}_i, \ell_i)\}_{1\leq i\leq N}$ representative of C *classes*, i.e., C call types in our case, where $\boldsymbol{s}_i$ is the i-th feature vector in the training set and ℓ_i is the corresponding class or label of $\boldsymbol{s}_i$, *e.g.*, $\ell_5 = 3$ means that the fifth element of the *training set* belongs to the third class. This training set is used to determine a map $f(\cdot|\{(\boldsymbol{s}_i, \ell_i)\}_{1\leq i\leq N})$ that infers a label from a given feature vector.

The map f is either learned on the training set by minimizing a loss function representing the cost paid for inaccuracy of predictions (i.e., discrepancy between the predicted and the actual label) or derived from a prior choice of a *similarity measure* that compares new test data to training examples. Neural network-based classifiers typically implement the first approach, whereas methods such as banks of matched-filters [37] or spectrogram correlators [32, 38] implement the second one.

As discussed below, our method relies on the second approach. This choice is mainly motivated by the will to build a robust and modular method where the similarity measure does not depend on the training set or on the number of call classes. It is also desirable to avoid using too many ("no-so-easy-to-tune") hyperparameters so as to ease the deployment of the method.

In the sequel, $\{\boldsymbol{s}_k : k > N\}$ stands for the *test* feature vectors that the system must classify. Given such a test feature vector $\boldsymbol{s}_k$ with $k > N$, $\widehat{\ell}_k = f(\boldsymbol{s}_k | \{(\boldsymbol{s}_i, \ell_i)\}_{1 \leq i \leq N})$ is the output label in $\{1, 2, \ldots, C\}$ assigned to $\boldsymbol{s}_k$.

In the method proposed below, feature vectors are digitized time-series of calls. It is assumed that detection of regions of interest within the time-series has already been achieved either automatically or manually. In Sections II A and II B, the sparse representation and classification framework for calls is presented. Sections II C and II D introduce the compression and the rejection options. In Section II E, an overall description of the procedure is given.

A. From standard similarity measures to sparse representation

There exists a wide variety of similarity measures, *e.g.* Euclidean distance, absolute value, likelihood, correlation, etc. For instance, let $|\langle \boldsymbol{s}_k, \boldsymbol{s}_i \rangle|$ be the non negative normalized scalar product or *correlation* between a signal $\boldsymbol{s}_k$ and a signal $\boldsymbol{s}_i$. For approaches such as banks of matched filters or spectrogram correlators, the map f chooses the class that maximizes the correlation between a test signal $\boldsymbol{s}_k$, $k > N$, and all the signals in the training dataset, i.e.,

$$\hat{\ell}_k = \ell_{i^*}, \tag{1}$$

where $i^* = \operatorname{argmax}_{i \in \{0,1\ldots,N-1\}} |\langle \boldsymbol{s}_k, \boldsymbol{s}_i \rangle|$.

A well-known extension of such an approach is the K Nearest Neighbors algorithm (KNN) [39] where $\boldsymbol{s}_k$ is assigned to the most common class among its K nearest neighbors (*e.g.*, the K signals in the training dataset having the highest correlation with $\boldsymbol{s}_k$). In general, choosing K greater than one is beneficial as it reduces the overall noise [40] .

Beyond KNN, the classification can be based on a similarity measure between the test signal $\boldsymbol{s}_k$ to be labeled and a *linear combination* of the K signals closest to $\boldsymbol{s}_k$. All training signals then become elementary *atoms* which can be combined to create new signals. In this way, the new representation space makes it possible to cover a larger space than the original training dataset and, as such, is expected to better capture the intrinsic/proper structure of the signals of interest. On the one hand, K should be small enough to prevent overfitting, especially in presence of noise. On the other hand, given a test signal, the similarity measure must help select a linear combination of atoms from the same class as the signal to guarantee a meaningful comparison between this one and each average model of each class. Therefore, the choice of K results from a trade-off between the risk of overfitting and the necessity to approximate sufficiently well the test signal.

Formally, it is assumed that any test signal $\boldsymbol{s}_k$ with dimension n from class c approximately lies in the linear span of the training signals associated with this class, i.e.,

$$\boldsymbol{s}_k \approx \boldsymbol{A}_c \boldsymbol{w}_c, \text{ with } \|\boldsymbol{w}_c\|_0 \leq K \ll N_c, \tag{2}$$

where $\boldsymbol{A}_c \in \mathbb{R}^{n \times N_c}$ is a matrix containing all the N_c training signals of length n belonging to the class c, $\boldsymbol{w}_c \in \mathbb{R}^{N_c}$ is a vector of weights used in the linear combination and $\|\boldsymbol{w_c}\|_0$ denotes the ℓ_0-pseudonorm that returns the number of non-zero coefficients in $\boldsymbol{w}_c$. When $\boldsymbol{s}_k$ can be represented by a small number of non-zero coefficients in the basis $\boldsymbol{A}_c$, model (2) is referred to as "sparse representation" in the signal processing literature [35]. The inequality $\|\boldsymbol{w}_c\|_0 \leq K$ is called the sparsity constraint. This constraint K is directly related to the "complexity" of each single call to be classified. Signals combining variability and high complexity (such as erratic signals) must be constructed from a large number of atoms while signals of low complexity should be composed of a few atoms. For instance, D calls of blue whales [41] are frequency-modulated (FM) sweep that could well be approximated by a linear combination of a few atoms. However, such calls exhibit variability in initial frequency, FM rate, duration, and bandwidth. Therefore, the ℓ_0 norm of $\boldsymbol{w}_c$ is small for each single call but the active atoms, corresponding to non-zero entries of $\boldsymbol{w}_c$, can be different from one call to another so that N_c must be large. Note that model (2) is an approximation as calls may be affected by local propagation conditions and noise. However, the very good results obtained in Sec. III indicate that it is sufficiently accurate for classification purposes. Examples of test signal reconstruction with training signals are shown in the appendix for real calls.

B. Sparse Representation-based Classification

Based on a linear model similar to (2), Wright et al. proposed a Sparse Representation-based Classifier (SRC) in [34]. It achieved impressive results in a wide range of applications such as bird classification [42], EEG signal classification [43], face recognition [34, 44]. Originally applied to face recognition, we suggest adapting this approach to our context. To this end, this subsection recalls the SRC procedure, whereas the next two propose additional features to improve SRC performance in our particular application.

SRC assumes that test signals can be represented by a linear combination of training signals. In our context, these signals are digitized time-series and represent the input feature vectors of the classifier. SRC is a two-step procedure: (i) it seeks the linear combination of training signals that best approximates — in the sparse sense — the test signal and (ii) chooses the class that mostly contributes to this approximation. More precisely, the true label of the test signal $\boldsymbol{s}_k$ being unknown, $\boldsymbol{s}_k$ is first represented as a linear combination of all training signals stored in a matrix $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \cdots, \boldsymbol{A}_C] \in \mathbb{R}^{n \times \sum_{c=1}^{C} N_c}$, where C is the number of call classes, i.e.,

$$\boldsymbol{s}_k \approx \boldsymbol{A}\boldsymbol{w}, \text{ with } \|\boldsymbol{w}\|_0 \leq K. \tag{3}$$

Ideally, the entries of $\boldsymbol{w} \in \mathbb{R}^{\sum_{c=1}^{C} N_c}$ are all zeros except at most K entries related to the training signals from the same class as the test signal. For instance, if $\boldsymbol{s}_k$ belongs to class c, i.e., $\ell_k = c$, then $\boldsymbol{w}$ should ideally satisfy $\boldsymbol{w} = [0, \cdots, 0, \boldsymbol{w}_c^T, 0, \cdots, 0]^T$ where $\boldsymbol{w}_c \in \mathbb{R}^{N_c}$ and $\|\boldsymbol{w}_c\|_0 \leq K$. Therefore, the actual class of the test signal could be obtained by estimating $\boldsymbol{w}$ and finding the indexes of the nonzero entries of $\boldsymbol{w}$. However, in practice, because of the noise and the non-orthogonality between training signals from different classes, nonzero entries of $\boldsymbol{w}$ may appear at indexes not related to the true class of the test signal. Consequently, the class label for the test signal is not determined by finding the indexes of the nonzero entries of $\boldsymbol{w}$ but by finding the class-specific entries of $\boldsymbol{w}$ yielding the best approximation of $\boldsymbol{s}_k$ in (3).

More specifically, the two-step procedure of SRC is as follows:

1. Estimate $\boldsymbol{w}$ by sparsely encoding $\boldsymbol{s}_k$ over the basis $\boldsymbol{A}$. i.e., by solving

$$\boldsymbol{w}^* = \underset{w}{\operatorname{argmin}} \|\boldsymbol{s}_k - \boldsymbol{A}\boldsymbol{w}\|_2^2, \text{ with } \|\boldsymbol{w}\|_0 \leq K. \tag{4}$$

   Sparse encoding can be performed with pursuit algorithms [35] or ℓ_1-norm minimization [45]. In Section III, this step is implemented with orthogonal matching pursuit (OMP) [46].

2. Associate $\boldsymbol{s}_k$ to the class $\hat{\ell}_k$ that satisfies

$$\hat{\ell}_k = \underset{1 \leq c \leq C}{\operatorname{argmin}} \|\boldsymbol{s}_k - \boldsymbol{A}\delta_c(\boldsymbol{w}^*)\|_2^2, \tag{5}$$

   where $\delta_c(\boldsymbol{w}^*)$ is a characteristic function that selects the coefficients of $\boldsymbol{w}^*$ associated with the c-th class. For any $\boldsymbol{w} \in \mathbb{R}^{\sum_{c=1}^{C} N_c}$, $\delta_c(\boldsymbol{w}) \in \mathbb{R}^{\sum_{c=1}^{C} N_c}$ is a vector whose nonzero entries are the entries in $\boldsymbol{w}$ that are related to the c-th class. For instance, if $\boldsymbol{w} = [\boldsymbol{w}_1^T, \boldsymbol{w}_2^T, \cdots, \boldsymbol{w}_C^T]^T$ where each $\boldsymbol{w}_i$ belongs to class i, then $\delta_c(\boldsymbol{w}) = [0, \cdots, 0, \boldsymbol{w}_c^T, 0, \cdots, 0]^T$. The solution to (5) is found by exhaustive search through all the classes.

C. Compression option

Ideally, the training dataset $\boldsymbol{A}$ should span the space that includes any mysticete call we wish to classify. In particular, for each class, $\boldsymbol{A}_c$ should incorporate enough variability to model all possible calls of the same class. It is thus desirable to inject in $\boldsymbol{A}$ the maximum amount of information we have on these calls. However, the computational complexity of (4) grows with the size of $\boldsymbol{A}$ without necessarily adding any performance improvement if $\boldsymbol{A}$ contains redundant signals. To limit redundancy in $\boldsymbol{A}$ and thus achieve a trade-off between variability and computational load, we suggest building a lower dimensional dictionary $\boldsymbol{D} = [\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_C]$ from the training dataset, where each submatrix $\boldsymbol{D}_c$ has $N_c' \leq N_c$ columns, i.e., $\boldsymbol{D}_c \in \mathbb{R}^{n \times N_c'}$. Each $\boldsymbol{D}_c$ is found as the subdictionary that leads to the best possible representation for each training signal of class c with the sparsity constraint (4). More precisely, the new subdictionary $\boldsymbol{D}_c$ for class c is derived by solving the minimization problem:

$$\begin{aligned} &\min_{\boldsymbol{D}_c, \boldsymbol{W}} \|\boldsymbol{A}_c - \boldsymbol{D}_c \boldsymbol{W}\|_F^2 \\ &\text{subject to } \|\boldsymbol{w}_i\|_0 \leq K, \, \forall\, 1 \leq i \leq N_c, \end{aligned} \tag{6}$$

where $\boldsymbol{W} = [\boldsymbol{w}_1, \cdots, \boldsymbol{w}_{N_c}]$ and $\boldsymbol{w}_i \in \mathbb{R}^{N_c'}$. The minimization problem (6) is commonly referred to as "dictionary learning" and is only performed offline once. Numerical solutions to (6) can be obtained with the method of optimized direction (MOD) [47], K-SVD [48] or online learning [45]. Once the lower dimensional dictionary is learned, $\boldsymbol{A}$ and $\boldsymbol{A}_c$ are replaced by $\boldsymbol{D}$ and $\boldsymbol{D}_c$ in (4) and (5), respectively and $\delta_c(\cdot)$ is adapted to the size of $\boldsymbol{D}$. In addition to removing the redundant information in the learning process, dictionary learning extracts the salient feature of $\boldsymbol{A}$ and this thus expected to limit the sensitivity to noisy training signals or to overfitting issues.

D. Rejection option

A major challenge in automatic classification of underwater sounds is the management of "noise". In our context, noise is defined as any test signal, fed into the classifier, that does not belong to one of the C output mysticete call classes of the classifier. This noise can be:

- Transient noise or interference that designates any transient signal of no interest for the classifier, *e.g.* calls of other whales, ship noise, airguns, earthquakes, ice tremors, etc.
- Background noise which is a mixture of numerous unidentifiable ambient sound sources that does not include any transient signal.

The rejection option offers the capability of refusing to assign the examined signal to any class, possibly prompting for a deeper investigation by a human analyst. In [34, Sec. 2.4], a rejection option is proposed for SRC. It relies on the assumption that a valid test signal has a sparse representation whose nonzero entries concentrate mostly on one class, whereas a signal to be rejected has coefficients spread widely among multiple classes. While such an assumption may be valid in applications such as face recognition [34], it is not applicable in our context. The main reason is that transient underwater acoustic noises may have a non-negligible amount of their energy lying in a subspace in which a specific class of calls resides. For instance, the sparse coefficients of impulsive noise are likely to concentrate on classes related to impulsive calls (such as the fin whale 20 Hz calls presented in Sec. III A), whereas tonal noise coefficients will be related to tonal calls having similar frequencies. To deal with noise, we propose to apply a post-processing procedure that decides whether the test signal actually lies in the subspace spanned by the column of the subdictionary corresponding to the class chosen by SRC. More precisely, the result of SRC is validated if the estimated Signal-to-Interference-plus-Noise Ratio (SINR)

$$\text{SINR}(\boldsymbol{s}_k, \hat{\ell}_k) = \frac{||\boldsymbol{D}\delta_{\hat{\ell}_k}(\boldsymbol{w}^*)||_2^2}{||\boldsymbol{s}_k - \boldsymbol{D}\delta_{\hat{\ell}_k}(\boldsymbol{w}^*)||_2^2} \tag{7}$$

is greater than some threshold. Based on model (2), $\boldsymbol{D}\delta_{\hat{\ell}_k}(\boldsymbol{w}^*)$ is an estimate of the signal of interest and $\boldsymbol{s}_k - \boldsymbol{D}\delta_{\hat{\ell}_k}(\boldsymbol{w}^*)$ is an estimate of the interference plus background noise. This criterion measures the reconstruction quality of the test signal $\boldsymbol{s}_k$ when approximated by a linear combination of the elements of $\boldsymbol{D}_{\hat{\ell}_k}$. It is inspired by Constant False Alarm Rate (CFAR) detectors of known signal in noise with unknown power, which show optimal properties with respect to detection performance [22, 23, 49]. The methodology used to set the SINR threshold is presented in Sec. III C 2. A key aspect of our approach is that the classifier does not need to learn features of transient noises to reject them. This differs from methods such as [25] where noise features are learned by neural networks or from [24] where, for each class of noise, "*a parametric model of noise is introduced. The models are based on the spectral properties of typical kinds of impulsive noise observed in the data*" [24, pp. 360]. This implies to find exhaustive examples of underwater noise, which seems difficult given the complexity of the underwater environment. The characteristics of sensed underwater sounds are highly dependent on the anthropogenic, biological, geological or oceanographic environment as well as on the way sensors are mounted in the water column. So the noise learned or modeled in one context can hardly be transposed to another one.

E. Overall procedure

The classification process resulting from the foregoing considerations is hereafter referred to as SINR-SRC. It is summarized as follows and illustrated with two classes in Figure 1.

1. Offline selection of training signals representative of their call class.

2. Offline application of the compression option (6) if required.

3. Given some test signal $\boldsymbol{s}_k$, perform a sparse encoding of $\boldsymbol{s}_k$ over dictionary $\boldsymbol{D}$ by computing:

$$\boldsymbol{w}^* = \underset{w}{\operatorname{argmin}} \|\boldsymbol{s}_k - \boldsymbol{D}\boldsymbol{w}\|_2^2, \text{ with } \|\boldsymbol{w}\|_0 \leq K.$$

4. Application of SRC by computing the class contributing most to the test signal $\boldsymbol{s}_k$:

$$\hat{\ell}_k = \underset{1 \leq c \leq C}{\operatorname{argmin}} \|\boldsymbol{s}_k - \boldsymbol{D}_c \delta_c(\boldsymbol{w}^*)\|_2^2.$$

5. Application of the rejection option: if $\mathrm{SINR}(\boldsymbol{s}_k, \hat{\ell}_k)$ is greater than some threshold, the result provided by SRC is validated, otherwise $\boldsymbol{s}_k$ is considered as noise.

This SINR-SRC procedure can be illustrated by the scheme shown in Fig. 1.

In addition to the good classification performance achieved by SINR-SRC (see Sec. III), note also that it is modular, which can be very useful in an operational context. For instance, if a new class of mysticete calls must be added to an existing SINR-SRC classifier, there is no need to "retrain" the entire classifier as required in approaches such as neural networks, random forest or support vector machine. Only the new subdictionary associated to the new class must be learned. Moreover, to reduce miss-classifications of online passive acoustic monitoring, prior information such as the geographical position of the sensor could be taken into account by removing the sub-dictionaries in $\boldsymbol{D}$ corresponding to species whose habitats are known to be far away from the sensor.

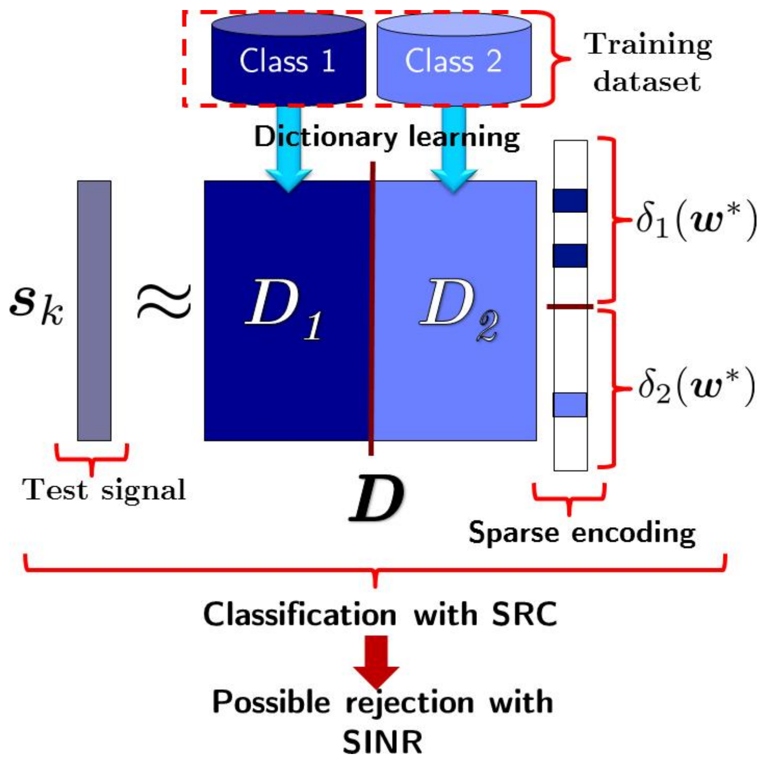

Figure 1. Overview of the classification method for 2 classes.

III. EXPERIMENTAL RESULTS

A. Call library

SINR-SRC is evaluated for five call types: Antarctic blue whale Z-calls [50, 51], two types of Madagascar pygmy blue whale calls [50], fin whale 20 Hz calls [52], North-Pacific blue whale D-calls [26, 41]. These calls have been chosen because:

- They all overlap in frequencies and some of them have similar durations so they cannot be

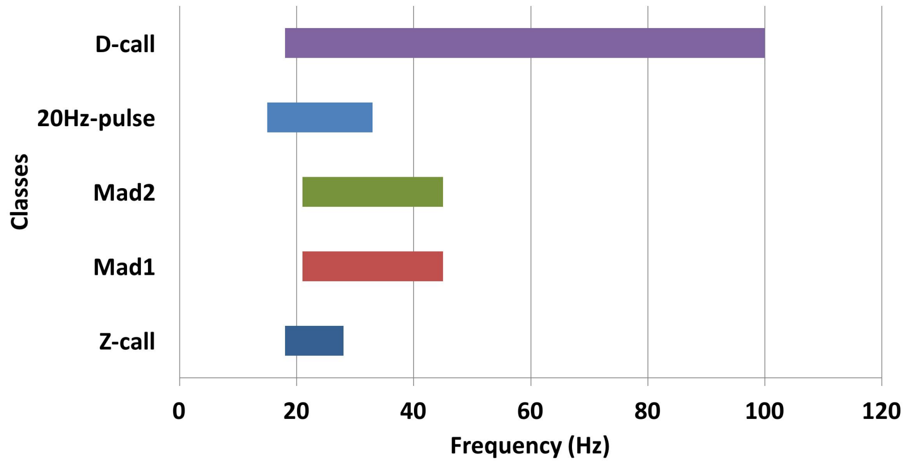

Figure 2. Frequency range of each call type.

discriminated based on these two elementary features (Fig. 2 and 3).

- They offer some variety in terms of signal types: pulsive, tonal sounds or frequency-modulated (FM) sweeps (Fig. 4).

- They exhibit different levels of variability: from almost stereotyped (*e.g.*, Z-calls) to variable in duration, bandwidth and FM rate (*e.g.*, D-calls).

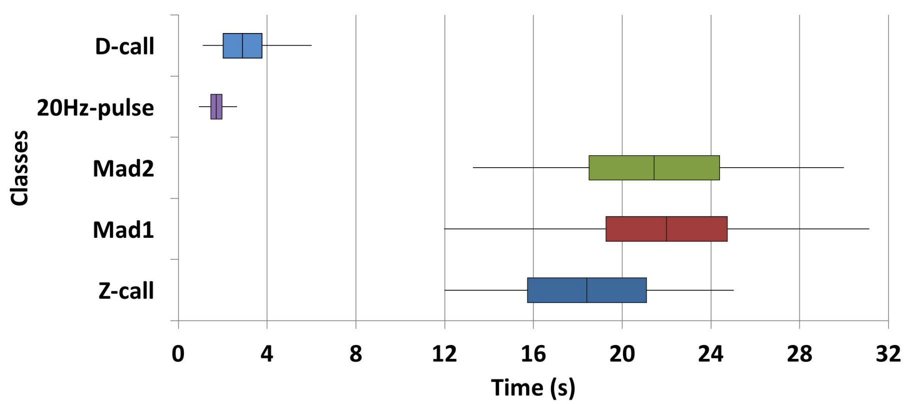

Figure 3. Boxplot of durations for each call type.

The five call types were manually extracted from three datasets.

The DEFLOHYDRO dataset: Three autonomous hydrophones were deployed near the French territories in the Southern Indian Ocean from October 2006 to January and April 2008. The objective of the project was to monitor low-frequency acoustic signals, including those produced by large whales [53]. The three instruments were widely spaced and located in the Madagascar

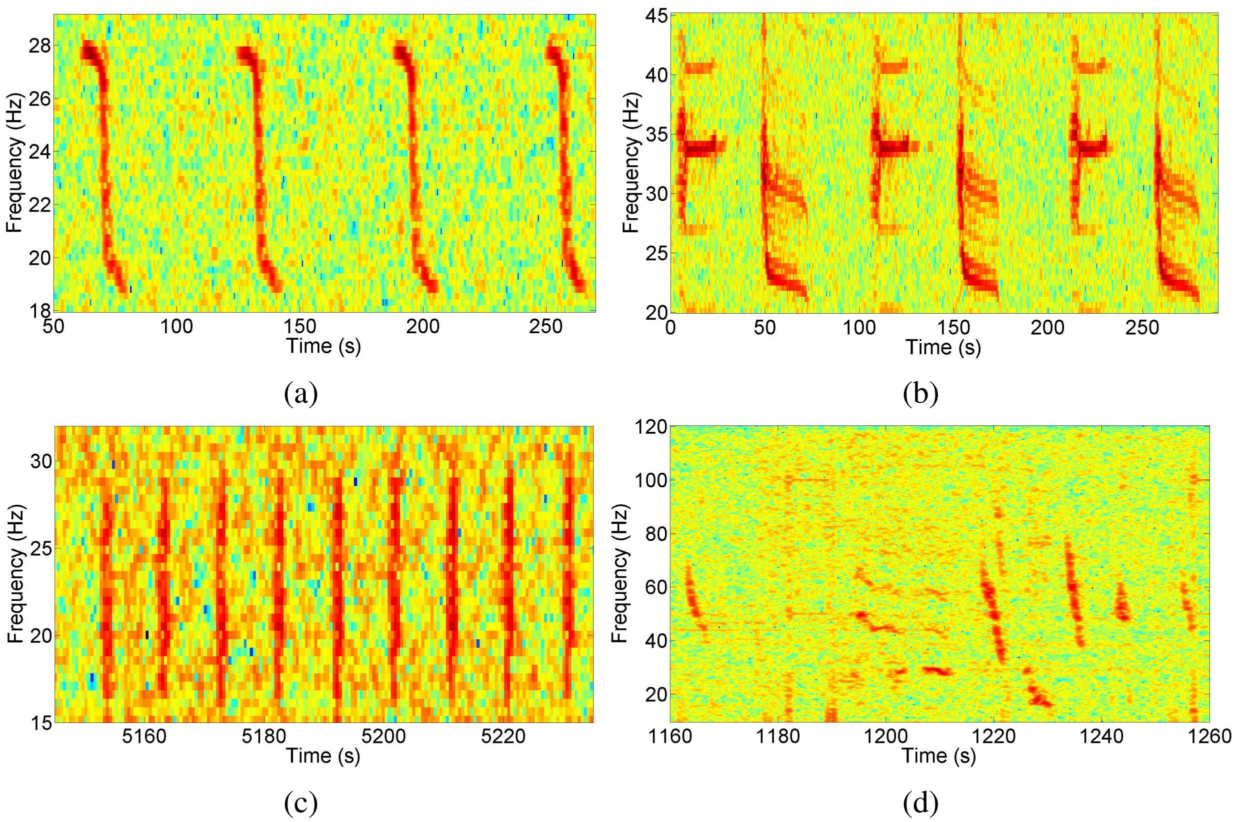

Figure 4. Examples of spectrograms from the call library. (a) four Z-calls produced by Antarctic blue whales, (b) two types of alternative calls produced by Madagascar pygmy blue whales, (c) 20 Hz pulse train produced by fin whales, (d) five D-calls produced by North-Pacific blue whales.

Basin, about 320 nautical miles (nm) south of La Reunion Island, and 470 nm to the northeast (NEAMS) and 350 nm to the southwest (SWAMS) of Amsterdam Island. The mooring lines were anchored on the seafloor between 3410 and 5220 m depths and the hydrophones were deployed near the sound channel axis (SOFAR) between 1000 m and 1300 m. The instruments recorded sounds continuously at a sampling rate of 250 Hz (frequency range 0.1-110 Hz) [50]. 254 Z-calls and 1000 fin whale 20 Hz calls were manually extracted from this dataset.

The OHASISBIO dataset: In continuation to the DEFLOHYDRO experiment, a network of hydrophones was initially deployed in December 2009 at five sites in the Southern Indian Ocean. This experiment was designed to monitor low-frequency sounds, produced by seismic and volcanic events, and by large baleen whales [17, 54]. 551 Madagascar pygmy blue whale calls were manually extracted from the data recorded by La Reunion Island hydrophone in the Madagascar Basin (geographic coordinates : +26° 05' S, +058 °08' E) in May 2015. 264 were type-1 calls and 287 were type-2, see Fig. 4.

The DCLDE 2015 dataset: These data have been obtained with high-frequency acoustic recording packages deployed in the Southern California Bight. 380 D-calls were extracted from data recorded at the CINMS B site (latitude: +34° 17' N, longitude: +120° 01' 7" W) in summer 2012 [26].

The whole library is composed of 2185 mysticete calls. Each call has been manually annotated in time and frequency: start and end time are identified as well as lowest and highest frequency of each call. All calls are band-pass filtered according to their annotation and resampled at 250 Hz. To apply SRC, all calls must have the same number of time samples, which is easily achieved by zero-padding. As shown in Fig. 5, the library contains signals with a large variety of Signal-to-Noise Ratios (SNR). The SNR is here defined as the ratio of signal power to noise power, *measured in the frequency band of each individual call*.

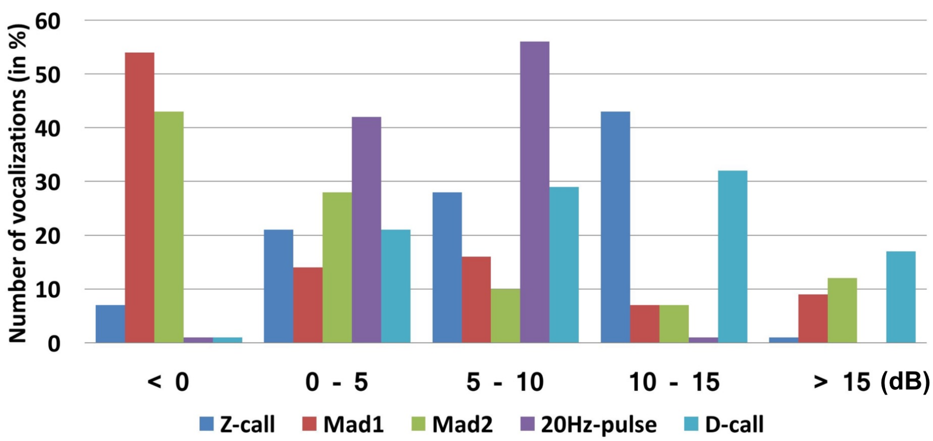

Figure 5. Distributions of the SNRs (in dB) of all the vocalizations in the dataset.

Note that four types of calls (Z-calls, 20Hz pulses, Mad1, Mad2) were recorded in the Indian ocean and one type (D-calls) in the Southern California Bight. Sensors of the OHASISBIO or DEFLOHYDRO networks can sense the first four types of calls in the same recordings [55] but North-Pacific blue whales D-calls are observed separately. In practice, this type of D-calls can therefore be differentiated from the other calls based on the assumed habitats. To challenge our method, location information was not taken into account. A similar approach was considered in [25]. In addition, blue whales in the Indian ocean also produce D-calls [56]. Although slightly different from D-calls of North-Pacific blue whales, these D-calls are also FM-like signals with variable initial frequency, FM rate, duration, and bandwidth. This suggests that our method could be relevant for these calls as well.

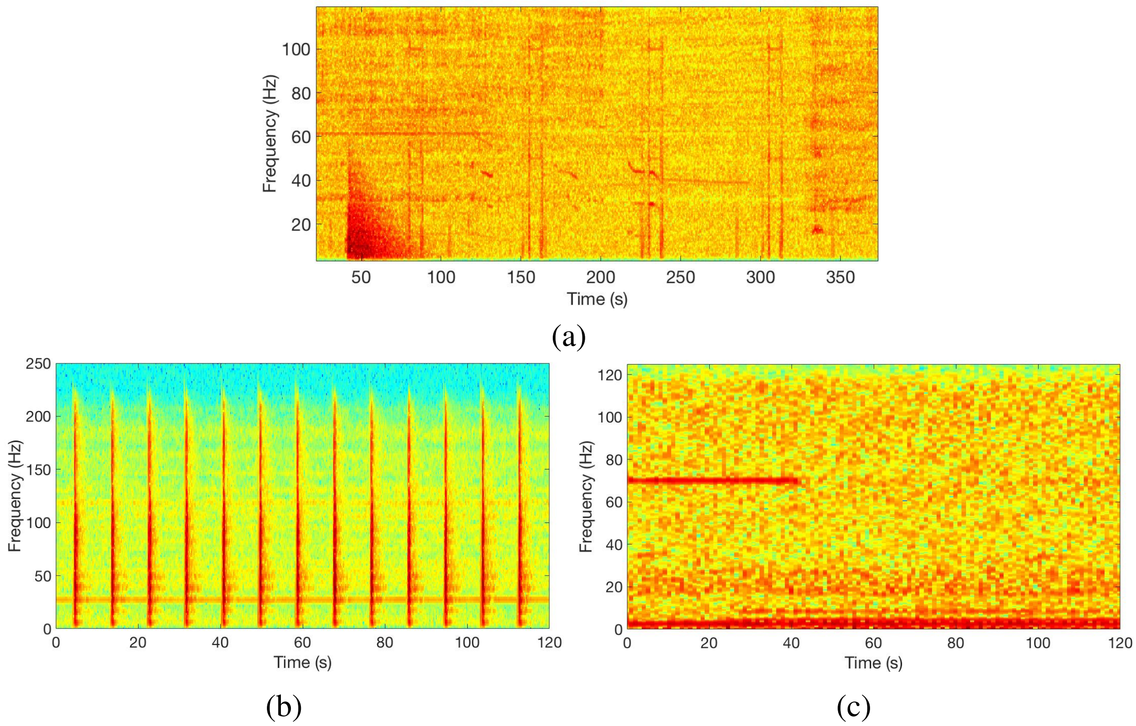

Figure 6. Examples of spectrograms from the noise library. (a) extracted from DCLDE 2015, (b) seismic survey noise provided by Sercel [57] and (c) oceanic noise extracted from DEFLOHYDRO.

B. Noise library

To test the robustness of SINR-SRC against noise, a noise library was also created. 5000 noise samples were extracted from the DEFLOHYDRO dataset, 5000 from the DCLDE 2015 dataset and 5000 more from a dataset, provided by Sercel [57], recorded during seismic surveys. The first 5000 noise samples mainly correspond to what is called "background noise" in Sec. II D and the others are mostly transient signals of no interest for the classifier, i.e., "interference" (see Fig. 6). In practice, the features (duration, bandwidth, power, etc.) of the noise samples injected into the classifier depends on the actual behavior of the detector used to identify the region of interest before classification. Since we would like to test the performance of our classifier irrespective of the detector, the noise samples were randomly extracted from the datasets. In addition, to challenge the method, noise samples were filtered so that their bandwidths and durations were chosen identical to bandwidths and durations of mysticete calls to be classified. This corresponds to a worst-case scenario for the classifier as filtered noise samples will have a greater amount of energy in the subspaces in which calls reside, leading to an increase of SINR (7).

C. Performance

The performance of SINR-SRC is first analyzed and compared with an implementation of a state-of-the-art method [29], in the absence of a rejection option. Results with the rejection option activated are then presented. The impact of the dictionary size as well as the sparsity constraint is discussed at the end of this section. The performance of the classifier is measured using cross-validation. As shown in Table I, for each class (with the exception of noise), 100 calls are randomly selected for training and the remaining calls in this class are used for testing. All the tests presented are averaged over 100 random selections of the training set to ensure that the results and conclusions do not depend on any specific choice of the training data. For each class, the recall metric, used below, is defined as the ratio between calls correctly classified and the total number of call in this class. This metric is sometimes referred to as sensitivity or true positive rate. A recall of 100% for Z-calls class means that all Z-calls have been correctly classified.

1. Results without rejection

Table II shows the average confusion matrix of the SRC algorithm without rejection and without injecting noise in the classifier. Each column of the matrix represents the percentage of calls in a predicted class while each row represents the percentage of calls in an actual class. The standard deviation of the classification results is also displayed in Table II. For this test, no reduction of the dictionary dimension is applied, i.e., $\boldsymbol{D} = \boldsymbol{A}$ and the sparsity constraint K is set to 3 (impact of these parameters on the classification performance is discussed in Sec. III C 2). An overall average recall of 99% is obtained. The SRC classifier not only makes very few errors but is also robust to training dataset changes.

For comparison, Table III displays the classification results obtained with an implementation of the time-frequency based method introduced in [29]. Similarly to SINR-SRC, this method is modular and is endowed with a rejection option that requires no noise training. It relies on the extraction of four amplitude-weighted time-frequency attributes: the average frequency, the frequency variation, the time variation, and the slope of the pitch track in time-frequency space. In our implementation inspired by [29], this extraction is performed on several spectrograms, each spectrogram being tuned to the time-frequency features of a specific class. The attributes extracted from each spectrogram are aggregated and then used as inputs of a quadratic discriminant function analysis classifier. This method yields slightly worse performance than SINR-SRC (without rejection option). Its average recall is 92.36% compared to 99.46% for SINR-SRC. Note also that SINR-SRC provides much smaller standard deviations. The method inspired by [29] learns an average model for each call class and is therefore strongly dependent on the quality of the training calls. When the training database contains no "outliers", the resulting model is accurate and leads to good classification results. However, in presence of a few calls with poor quality, the model is affected and the performance of such a method decreases. In contrast, the dictionary of SINR-SRC involves sufficiently many atoms so that the reconstruction of the test signal is always good enough to yield good classification performance.

2. *Results with the rejection option activated*

We now illustrate the performance of SINR-SRC when the rejection option is activated. We recall that, as opposed to alternative methods such as [24, 25], rejection of noise is achieved without learning or modeling noise features, i.e., no dictionary is built from noise data. An input is rejected by the classifier if the estimated SINR, obtained by computing (7), is lower than some threshold. This approach is very efficient to discriminate noise data from calls of interest [23]. There exists numerous ways of setting the rejection threshold. For instance, it can be empirically chosen by the user according to the context and based on his own experience or it can rely on performance statistics.

For instance, we hereafter present a method that is based on the estimation of a false-alarm probability as commonly done in the Neyman-Pearson framework for binary hypothesis testing. Assuming that the probability density function (pdf) of the SINR metric is known when noise samples are injected into the classifier, a rejection threshold guaranteeing a user-specified false-alarm probability can then be found. However, since the space of all possible underwater transient noises is very large, it is hardly possible to know precisely this pdf in practice. Therefore, we resort to an empirical approach and inject into the classifier synthetic random noise samples to obtain a pdf from which we can set a threshold. This noise is synthetic so as to be as independent as possible of a specific dataset. In our experiment, we generate independent and identically distributed samples following the standard Gaussian distribution. Any variance different from 0 could be used, as the SINR metric is scale invariant. The synthetic noise is then obtained by filtering these samples in time and frequency. The filters have bandwidths and durations identical to bandwidths and durations of mysticete calls to be classified. As explained in Sec. III B, this corresponds to a worst-case scenario for our method because such a noise will yield a greater SINR than noise with any other bandwidth and duration. In practice, actual detectors possibly used ahead of the classifier are unlikely to trigger the classifier with a false alarm signal whose bandwidth and duration exactly match those of an actual mysticete call. The consideration of worst-case scenarios is justified by the will to measure achievable classification performance irrespective of the detector. Rejection thresholds are estimated on each SINR distribution obtained after injecting Gaussian samples *into each dictionary*. Figure 7 shows an example of a rejection threshold chosen by setting a false-alarm probability at 1% on the SINR distribution obtained with filtered Gaussian samples injected into the Z-call dictionary. Note that distributions other than Gaussian may have been relevant to model noise samples. However, Fig. 7 indicates that the SINR distribution (in red) of real noises (not necessarily Gaussian) obtained after SRC is close to the distribution obtained with Gaussian input samples. Once again, the rejection threshold could be selected with alternative methods. It is beyond the scope of the paper to thoroughly investigate this point; we rather focus our attention on the general methodology and the classifier structure.

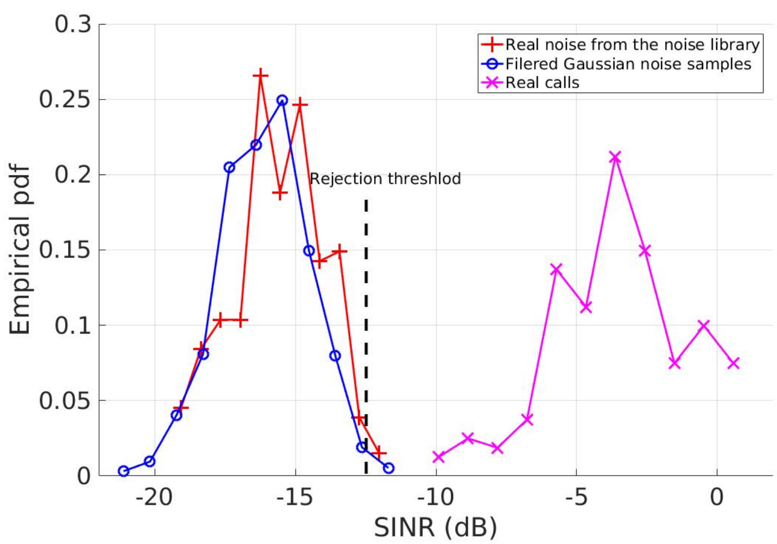

Figure 7. Distribution of SINR, as computed in (7), for Gaussian samples (in blue), real noise (in red) and real calls from the test dataset (in magenta), all identified as Z-calls according to the SRC algorithm without the rejection option. For a 1% false-alarm probability, the rejection threshold is set to -12.5 dB.

Table IV shows the average confusion matrix of the SINR-SRC algorithm with rejection. As expected, activating the rejection option yields a slight drop in the average recall. This drop is mostly significant for D-calls due to their high variability in duration, frequency range and energy distribution which cause that certain calls in the test dataset are considered as transient noise and therefore rejected. However, observe that 93.34% of noise inputs are correctly rejected. This clearly shows that SINR-SRC is capable of efficiently handling input data that are unknown to the classifier. This property is highly desirable in the low-frequency underwater environment where interfering sound sources can be very active. The classification results of SINR-SRC with the rejection option *deactivated* are shown in Table V when noise inputs only are injected into the classifier. It can be seen that noise inputs are spread among the 5 classes with a slightly higher probability for classes of calls embedding impulsive structures with a large frequency slope. This is explained by the large number of transient signals in the noise library.

For comparison, the classification results obtained with the method derived from [29], with its rejection option, are shown in Table VI. A test signal is rejected if the Mahalanobis distance between its feature vector and its assigned mean attribute vector exceeds 3. This rejection option does not significantly reduce the recall. However, the noise rejection proposed in [29] is not as effective as the SINR-SRC rejection option. Actually, Tables IV and VI show that 93.3% of noise samples are correctly rejected by SINR-SRC, whereas only 66.4% are rejected by [29]. For a deeper analysis of the rejection performance for SINR-SRC and [29], zooms on the receiver operating characteristics (ROC) curves are shown in Figures 8 and 9. Such a comparison is all the more relevant that the noise rejection is controlled by both methods via one parameter only that we made vary. For our implementation of [29], this parameter is the threshold on the Mahalanobis distance between a test signal feature vector and its assigned mean attribute. For SINR-SRC, this parameter is the false alarm probability we can specify to all the SINR distributions obtained after injections of filtered Gaussian noise samples into the dictionaries. Given a specified false alarm probability for SINR-SRC, or a specified threshold on the Mahalanobis distance for our implementation of [29], we calculated the actual false alarm rates and recalls obtained by each method in presence of real noise and calls. We remind the reader that filtered noise samples have similar bandwidths and durations as those of mysticete calls to be classified, which is the worst-case scenario for both methods.

These ROC curves highlight the better ability of SINR-SRC to reject noise compared to the reference method. In particular, the offset in Figure 8 indicates that filtered noise tends to have average time-frequency attributes close to learned attributes of calls, whatever the type of call. In the worst-case scenario we have considered, the method derived from [29] cannot provide a false alarm rate smaller than 5%. Note also the following facts. To begin with, the noise rejection rate of 66.41% reported in the confusion matrix of Table VI corresponds to a false alarm rate of 33.59%. The reader can then verify that the recall values of Table VI can be retrieved from the ROC curves of Figure 8. In the same way, given that a specified false alarm probability of 1% on the SINR distributions yielded an actual false alarm rate of 6.66% for SINR-SRC (equivalently, a noise rejection rejection rate of 93.34% for this method), the recall values displayed in Table IV can be obtained from Figure 9. The ROC curves of Figure 9 also emphasize the relevance of setting a false alarm probability of 1%, leading to an actual false alarm rate of 6.66%. This choice is seemingly a good trade-off between false alarm rate and recall, even for D-calls. Indeed, beyond this false alarm probability, increases in false alarm rates become more important than gains in recalls.

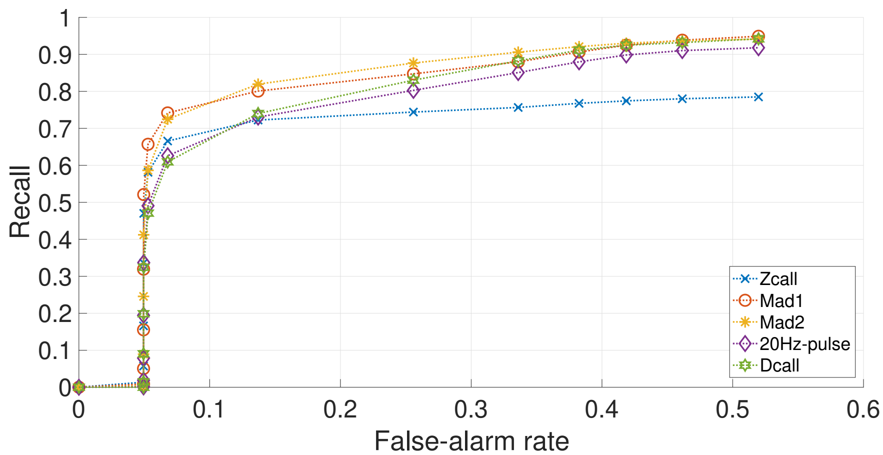

Figure 8. ROC curve for each class of the method derived from [29] with rejection option.

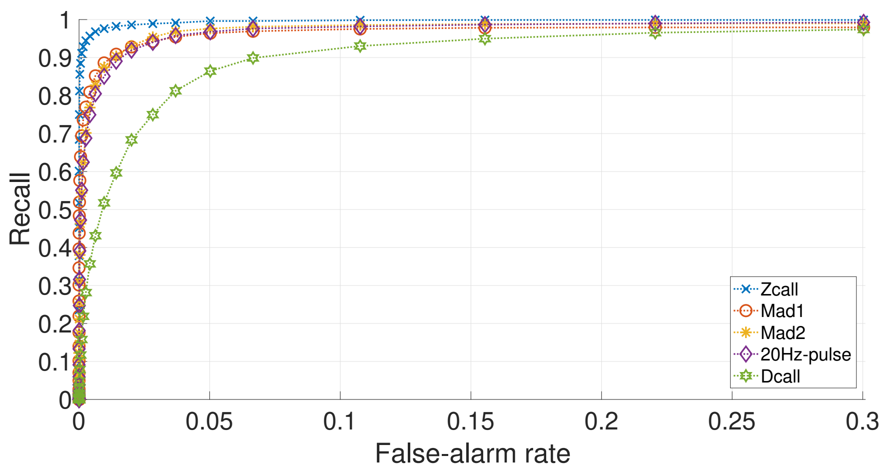

Figure 9. ROC curve for each class of SINR-SRC with rejection option.

So far, no reduction of the dictionary dimension has been considered, i.e., $\boldsymbol{D} = \boldsymbol{A}$. As mentioned in Sec. II B, limiting the redundancy by solving (6) during the training phase may be useful to reduce the computational complexity. Figure 10 shows the impact of the dictionary size N'_c on the classification performance for each call class. For this test, (6) was solved using online dictionary learning [45] (the Matlab code is available at `http://spams-devel.gforge.inria.fr/`). The dictionary size affects the recall and it is interesting to note that its impact is class-dependent. For stereotyped calls such as Z-calls, the size of the dictionary can be small since the dimension of the signal space is related to the call variability, which is low in this case. However, for varying signals such as D-calls, which also have overlapping features with 20 Hz-pulses, the classification recall increases (on average) with the dictionary size. In this experiment, choosing $N'_c = 40$ for each class is sufficient to achieve close-to-optimal performance.

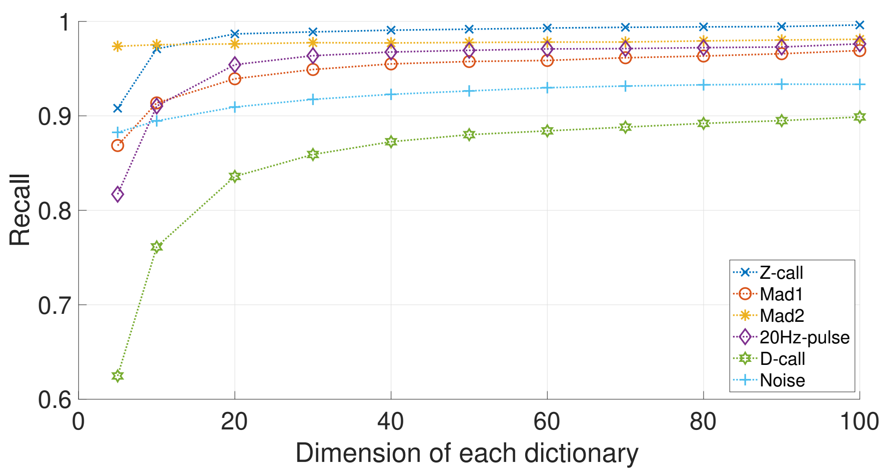

Figure 10. Average recall as a function of the dictionary size N'_c, $K = 3$ and rejection option activated.

The impact of the dictionary size on the computational complexity is visible in Figure 11 where the run-time-to-signal-duration ratio (RTSDR) of SINR-SRC is shown as a function of the dictionary size N'_c. This ratio is computed as the duration of the processing time divided by the total duration of the test dataset (58 h). SINR-SRC is implemented in Matlab (without parallel computing) and runs on a workstation with the 2.9 GHz Intel Core i7 processor, 8 Gio of RAM memory and a DDR3 internal hard drive. Most of the computation time is spent in solving (4) by using OMP, which makes the RTSDR increase with N'_c. In this experiment, the processing time increases linearly with N'_c. Therefore, according to Figure 10, the processing time can be divided by 2.5 by choosing $N'_c = 40$ instead of $N'_c = 100$ without any performance loss. For $N'_c = 40$, SINR-SRC took less than 24 seconds to process the 58 hours of tests signals, which meets the requirements of most PAM applications. Note that this time is expected to increase with the number of classes considered by the classifier.

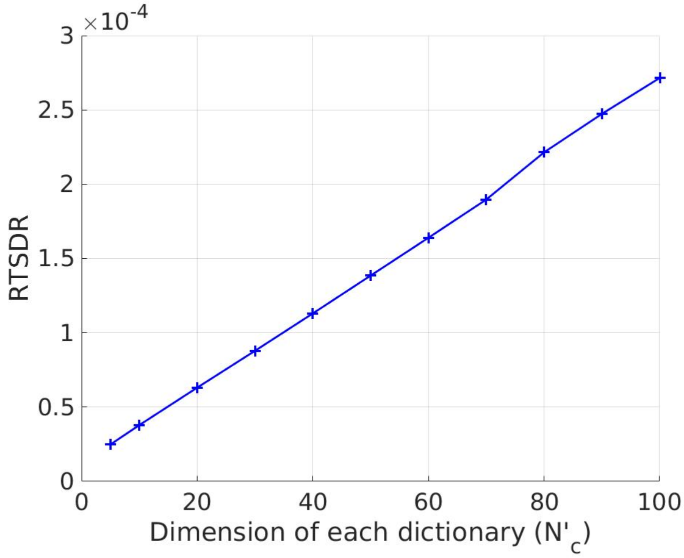

Figure 11. Run-time-to-signal-duration ratio as a function of the dictionary size N'_c.

As shown in Figure 12, the sparsity constraint K can also affect the classification recall. Similarly to the dictionary size, the optimal value for K depends on the variability and complexity of the test signals and is therefore class-dependent. However, no fine tuning is required. SINR-SRC performs better for all classes when K is greater than 1, $K = 1$ corresponding to a bank of matched-filters. For a sparsity constraint greater than 3 and less than 10, this test shows that SINR-SRC is robust to the choice of K. Since K contributes to the complexity of our algorithm, it may be relevant to limit it to 3 or 4 for the call classes tested in this experiment. In addition, choosing a large value for K (much greater than 10 for instance) may be detrimental to the classification performance as the SINR metric will tend to reject less noise samples [23, Sec. 4.1.2].

IV. CONCLUSION

Sparse representations have shown to be efficient to classify low frequency mysticete calls. Such representations model calls as linear combinations of atoms in an (overcomplete) dictionary in which many of the coefficients are zero. In this framework, the classifier seeks to approximate the input test signals with (a few) linear combinations of previously learned calls and assigns the class label that gives the best approximation. The proposed method directly processes the digitized time series and therefore does not suffer any loss of information due to a possible projection in

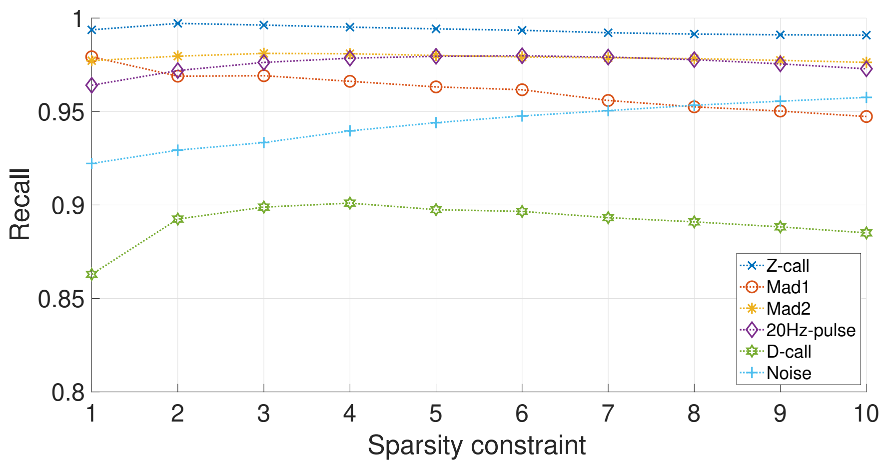

Figure 12. Average recall as a function of the sparsity constraint K, $N'_c = 100$ and rejection option activated.

another space (as can been done when extracting features from spectrograms or cepstrums). Since the classification is based on a measure of similarity, it relies on a few parameters, namely, the dictionary size and the sparsity constraint. These parameters reflect the degree of variability and complexity of a given call class. As shown in the numerical experiments, these parameters are easy to set and do not require a fine tuning.

Sparse representations also allows building simple confidence metrics to reject noise data. The SINR statistic (7) has been used at the output of the classifier and has rejected 93.3% of real noise data. With this approach, noise is handled without making the algorithm learn the features of real noise data. The overall method has been tested on five types of mysticete calls with overlapping time-frequency features and different degrees of variability. Numerical results have shown that, on the test dataset, 96.4% are correctly classified on average. As expected, stereotyped calls, such as Z-calls of Antarctic blue whale are easier to classify than more variable calls such as blue whale D calls, which can be incorrectly rejected by the SINR statistic.

Class labels can easily be removed or added to the proposed method. This can be useful for operational passive acoustic monitoring where prior information such as location of the sensor and/or time of the year can be taken into account to focus on specific species.

In a recent work [23], sparse representations have shown good performance for detecting mysticete calls. A possible extension of this work would therefore be to merge both approaches to jointly detect and classify mysticete sounds. Since calls are affected by local propagation conditions and noise, further work could also study the potential benefit of building dictionaries from parametric model of calls rather than/as well as from the call themselves. In addition, the SINR statistic could be used as a confidence metric (related to the threshold position) and also as a novelty detector. In this way, the SINR-SRC algorithm would not only offer the capability of rejecting noise but it could also be used to develop an automatic semi-supervised incremental learning algorithm able to build new dictionaries online. After detection by the SINR-SRC algorithm of an unknown structured signal, a human analyst could label it and decide to add it to a new dictionary for automatic classification of future occurrences of this new class of signals.

APPENDIX

Figures 13 and 14 show examples of Z and D-call reconstruction using Orthogonal Matching Pursuit (OMP) [46], with $K = 3$ atoms. These calls have been extracted from the DEFLOHYDRO and the DCLDE 2015 datasets described in Sec. III A.

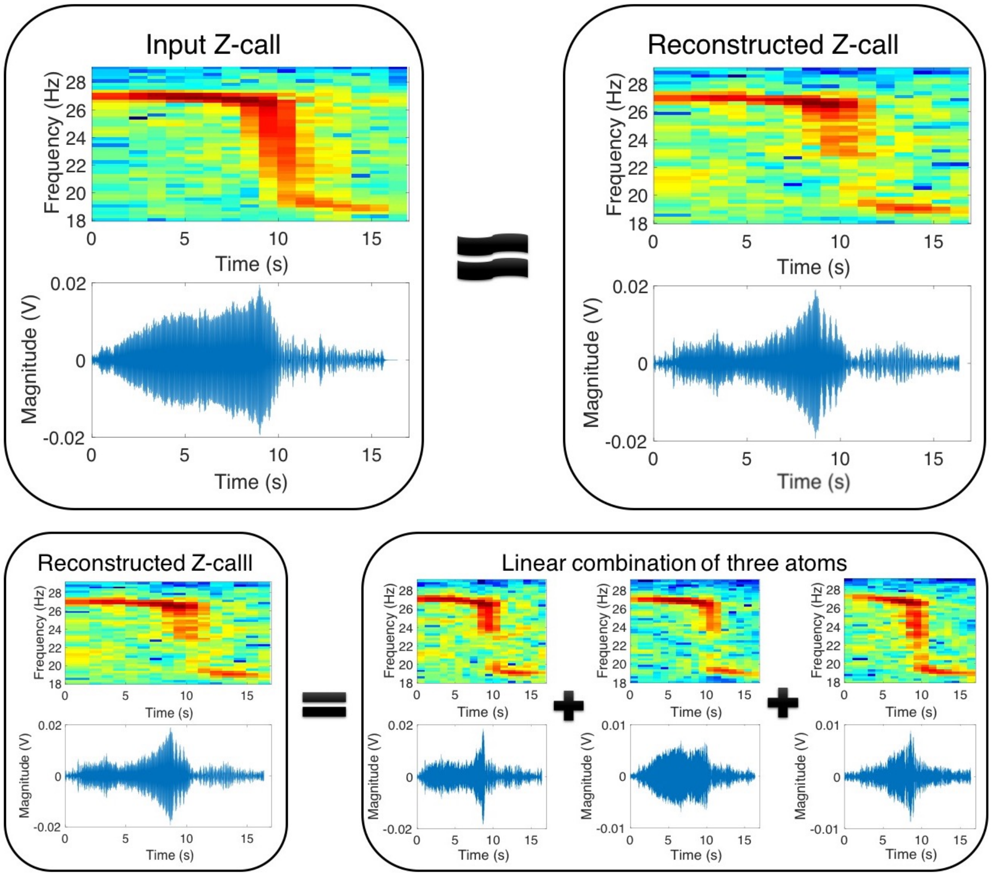

Figure 13. Example of Z-call reconstruction with OMP. The spectrogram representation and the temporal signal of a test Z-call are displayed on the top left. The spectrogram and time representations of the reconstructed signal with $K = 3$ are given on the top right. Below are the three atoms and their combination that provided the Z-call reconstruction.

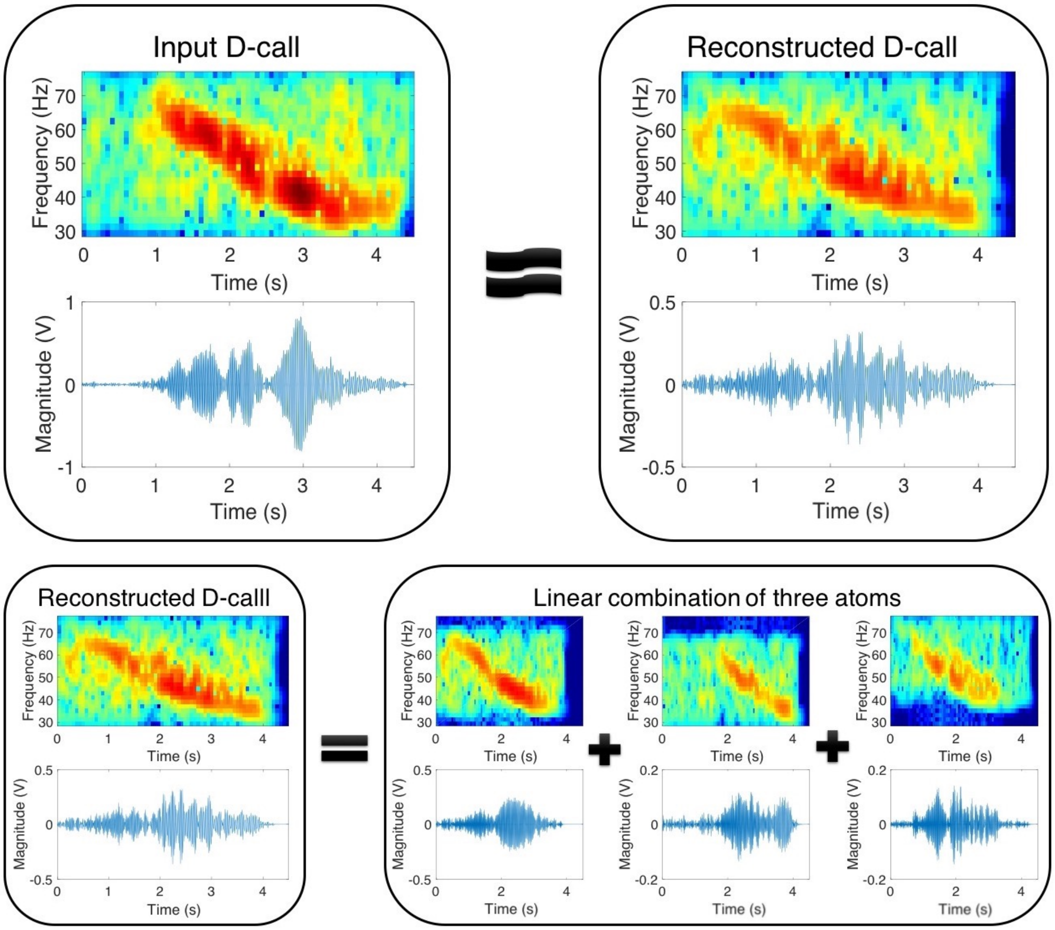

Figure 14. Example of D-call reconstruction with OMP. The spectrogram representation and the temporal signal of a test D-call are displayed on the top left. The spectrogram and time representations of the signal reconstructed by OMP with $K = 3$ are given on the top right. Below are the three atoms and their combination that provided the D-call reconstruction.

ACKNOWLEDGMENT

This work was funded by Sercel. The authors would like to thank Jean-Yves Royer of the University of Brest, CNRS Laboratoire Domaines Océaniques for providing the DEFLOHYDRO and the OHASISBIO datasets as well as Ana Širović and Simone Baumann-Pickering of the Scripps Institution of Oceanography for providing the DCLDE 2015 dataset.

The authors are also very thankful to the Associate Editor, Prof. Aaron Thode, whose invaluable comments and remarks made it possible to significantly improve this paper.

---

[1] D. K. Mellinger, K. M. Stafford, S. E. Moore, R. P. Dziak, and H. Matsumoto, "An overview of fixed passive acoustic observation methods for cetaceans," *Oceanography*, vol. 20, December 2007.

[2] S. E. Parks, C. W. Clark, and P. L. Tyack, "Short- and long-term changes in right whale calling behavior: The potential effects of noise on acoustic communication," *J. Acoust. Soc. Am.*, vol. 122, no. 6, pp. 3725–3731, 2007.

[3] A. M. Thode, K. H. Kim, S. B. Blackwell, C. R. Greene, C. S. Nation, T. L. McDonald, and A.M Macrander, "Automated detection and localization of bowhead whale sounds in the presence of seismic airgun surveys," *J. Acoust. Soc. Am.*, vol. 131, pp. 3726–3747, 2012.

[4] Renata S. Sousa-Lima, Thomas F. Norris, Julie N. Oswald, and Deborah P. Fernandes, "A review and Inventory of autonomous recorders fixed autonomous recorders for passive acouctic monitoring of marine mammals," *Aquat. Mamm.*, vol. 39, no. 1, pp. 21–28, 2013.

[5] David K. Mellinger, Stephen W. Martin, Ronald P. Morrissey, Len Thomas, and James J. Yosco, "A method for detecting whistles, moans, and other frequency contour sounds," *J. Acoust. Soc. Am.*, vol. 129, no. 6, pp. 4055–4061, 2011.

[6] Michael Bittle and Alec Duncan, "A review of current marine mammal detection and classification algorithms for use in automated passive acoustic monitoring," *Proc. Acoust.*, , no. November, 2013.

[7] W. M. X. Zimmer, *Passive Acoustic Monitoring of Cetaceans*, Cambridge University Press, 2011.

[8] M. A. McDonald, J. A. Hildebrand, and S. Mesnick, "Worldwide decline in tonal frequencies of blue whale songs," *Endangered Species Research*, vol. 9, no. 1, pp. 13–21, 2009.

[9] Tzu Hao Lin, Hsin Yi Yu, Chi Fang Chen, and Lien Siang Chou, "Automatic detection and classification of cetacean tonal sounds from a long-term marine observatory," *2013 IEEE Int. Underw. Technol. Symp. UT 2013*, 2013.

[10] Kathleen M. Stafford, Sue E. Moore, and Christopher G. Fox, "Diel variation in blue whale calls recorded in the eastern tropical Pacific," *Anim. Behav.*, vol. 69, no. 4, pp. 951–958, 2005.

[11] Paul O Thompson, Lloyd T Findley, and Omar Vidal, "20-Hz pulses and other vocalizations of fin whales, Balaenoptera physalus, in the Gulf of California, Mexico," *J. Acoust. Soc. Am.*, vol. 92, no. 6, pp. 3051–3057, 1992.

[12] David K. Mellinger, Carol D. Carson, and Christopher W. Clark, "Characteristics of Minke Whale

(Balaenoptera Acutorostrata) Pulse Trains Recorded Near Puerto Rico," *Mar. Mammal Sci.*, vol. 16, no. 4, pp. 739–756, 2000.

[13] Hui Ou, W W L Au, and Julie N Oswald, "A non-spectrogram-correlation method of automatically detecting minke whale boings.," *J. Acoust. Soc. Am.*, vol. 132, no. 4, pp. EL317—-22, 2012.

[14] Alison K. Stimpert, Whitlow W. L. Au, Susan E. Parks, Thomas Hurst, and David N. Wiley, "Common humpback whale (<i>Megaptera novaeangliae</i>) sound types for passive acoustic monitoring," *J. Acoust. Soc. Am.*, vol. 129, no. 1, pp. 476–482, 2011.

[15] S. E. Parks, A. Searby, A. Célérier, M. P. Johnson, D. P. Nowacek, and P. L. Tyack, "Sound production behavior of individual North Atlantic right whales: Implications for passive acoustic monitoring," *Endanger. Species Res.*, vol. 15, no. 1, pp. 63–76, 2011.

[16] Jason Gedamke, Daniel P. Costa, and Andy Dunstan, "Localization and visual verification of a complex minke whale vocalization," *J. Acoust. Soc. Am.*, vol. 109, no. 6, pp. 3038–3047, 2001.

[17] E. Leroy, F. Samaran, J. Bonnel, and J.-Y. Royer, "Seasonal and diel vocalization patterns of antarctic blue whale (balaenoptera musculus intermedia) in the southern indian ocean: A multi-year and multi-site study," *PloS one*, vol. 11, no. 11, pp. e0163587, 2016.

[18] Christopher W Clark, Robert Suydam, and Craig George, "Acoustic Monitoring of the Bowhead Spring Migration off Pt. Barrow, Alaska: Results from 2009 and Status of 2010 Field Effort," pp. 1–9, 2010.

[19] Tervo O.M., "Acoustic behaviour of bowhead whales Balaena mysticetus in Disko Bay, Western Greenland," *Tesis Dr.*, , no. April, pp. 138, 2011.

[20] Sm Wiggins, Ma McDonald, Lisa M Munger, Sue E Moore, and John Hildebrand, "Waveguide propagation allows range estimates for North-Pacific right whales in the Bering Sea," *Can. Acoust.*, vol. 32, no. 2, pp. 146–154, 2004.

[21] M. A. Roch, A. Širović, and S. Baumann-Pickering, "Detection, classification, and localization of cetaceans by groups at the scripps institution of oceanography and san diego state university (2003-2013)," *Detection, Classification, Localization of Marine Mammals using passive acoustics*, pp. 27–52, 2013.

[22] F.-X. Socheleau, E. Leroy, A. Carvallo Pecci, F. Samaran, J. Bonnel, and J.-Y. Royer, "Automated detection of antarctic blue whale calls," *J. Acoust. Soc. Am.*, vol. 138, no. 5, pp. 3105–3117, 2015.

[23] F.-X. Socheleau and F. Samaran, "Detection of mysticete calls: a sparse representation-based approach," *IMT Atlantique, Research report RR-2017-04-SC*, Oct. 2017. https://hal.

archives-ouvertes.fr/hal-01736178/document

[24] I. R. Urazghildiiev, C. W. Clark, T. P. Krein, and S. E. Parks, “Detection and recognition of north atlantic right whale contact calls in the presence of ambient noise,” *IEEE Journal of Oceanic Engineering*, vol. 34, no. 3, pp. 358–368, July 2009.

[25] X. C. Halkias, S. Paris, and H. Glotin, “Classification of mysticete sounds using machine learning techniques,” *J. Acoust. Soc. Am.*, vol. 134, no. 5, pp. 3496–3505, 2013.

[26] “Dclde 2015,” http://www.cetus.ucsd.edu/dclde/datasetDocumentation.html, Accessed: 2016-07-01.

[27] Carolyn M. Binder and Paul Hines, “Applying automatic aural classification to cetacean vocalizations,” *Proc. Meet. Acoust.*, vol. 17, pp. 070029, 2012.

[28] Federica Pace, “Automated classification of humpback whale (Megaptera novaeangliae) songs using Hidden Markov Models,” 2013.

[29] M. F. Baumgartner and S. E. Mussoline, “A generalized baleen whale call detection and classification system,” *J. Acoust. Soc. Am.*, vol. 139, pp. 2889–2902, 2011.

[30] X. Mouy, D. Leary, B. Martin, and M. Laurinolli, “A comparison of methods for the automatic classification of marine mammal vocalizations in the Arctic,” *2008 New Trends Environ. Monit. Using Passiv. Syst.*, 2008.

[31] Vasilis Trygonis, Edmund Gerstein, Jim Moir, and Stephen McCulloch, “Vocalization characteristics of North Atlantic right whale surface active groups in the calving habitat, southeastern United States,” *J. Acoust. Soc. Am.*, vol. 134, no. 6, pp. 4518–4531, 2013.

[32] D. K. Mellinger and C. W. Clark, “Recognizing transient low-frequency whale sounds by spectrogram correlation,” *J. Acoust. Soc. Am.*, vol. 107, pp. 3518–3529, 2000.

[33] Xavier Mouy, Mohammed Bahoura, and Yvan Simard, “Automatic recognition of fin and blue whale calls for real-time monitoring in the St. Lawrence,” *J. Acoust. Soc. Am.*, vol. 126, no. 6, pp. 2918–2928, 2009.

[34] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[35] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer Publishing Company, Incorporated, 1st edition, 2010.

[36] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*, Cambridge University

Press, 2012.

[37] I. R. Urazghildiiev and C. W. Clark, "Acoustic detection of north atlantic right whale contact calls using the generalized likelihood ratio test," *J. Acoust. Soc. Am.*, vol. 120, no. 4, pp. 1956–1963, 2006.

[38] XBAT, "eXtensible BioAcoustic Tool," www.birds.cornell.edu/brp/ (date last viewed 14/6/30), Cornell Laboratory of Ornithology, NY, U.S.A.

[39] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*, vol. 1, Springer series in statistics Springer, Berlin, 2001.

[40] T M Cover, and P E Hart, "Nearest Neighbor pattern classification," *IEEE Trans. Info. Theory*, vol. I, 1967.

[41] P. O. Thompson, L. T. Findley, O. Vidal, and W. C. Cummings, "Underwater sounds of blue whales, balaenoptera musculus, in the gulf of california, mexico," *Marine Mammal Science*, vol. 12, no. 2, pp. 288–293, 1996.

[42] Lee Ngee Tan, George Kossan, Martin L. Cody, Charles E. Taylor, and Abeer Alwan, "A sparse representation-based classifier for in-set bird phrase verification and classification with limited training data," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 763–767, 2013.

[43] Younghak Shin, Seungchan Lee, Minkyu Ahn, Hohyun Cho, Sung Chan Jun, and Heung No Lee, "Noise robustness analysis of sparse representation based classification method for non-stationary EEG signal classification," *Biomed. Signal Process. Control*, vol. 21, pp. 8–18, 2015.

[44] Enrique G. Ortiz, Alan Wright, and Mubarak Shah, "Face recognition in movie trailers via mean sequence sparse representation-based classification," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3531–3538, 2013.

[45] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Mar. 2010.

[46] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, Nov 1993, pp. 40–44 vol.1.

[47] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *Proceedings of the Acoustics, Speech, and Signal Processing, 1999. On 1999 IEEE International Conference - Volume 05*, Washington, DC, USA, 1999, pp. 2443–2446.

[48] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Sig. Proc.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[49] L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*, Addison-Wesley, pp. 1 - 524, Reading, Massachusetts, 1991.

[50] F. Samaran, K. M. Stafford, T. A. Branch, J. Gedamke, J.-Y. Royer, R. P. Dziak, and C. Guinet, "Seasonal and geographic variation of southern blue whale subspecies in the indian ocean," *PLoS ONE*, vol. 8, no. 8, pp. 1– 10, 2013.

[51] K. M. Stafford, D. R. Bohnenstiehl, M. Tolstoy, E. Chapp, D. K. Mellinger, and S. E. Moore, "Antarctic type blue whale calls recorded at low latitudes in the Indian and eastern Pacific Oceans," *Deep Sea Res., Part I*, vol. 51, pp. 1337–1346, 2004.

[52] A. Širović, J. A. Hildebrand, S. M. Wiggins, and D. Thiele, "Blue and fin whale acoustic presence around antarctica during 2003 and 2004," *Marine Mammal Science*, vol. 25, no. 1, 2009.

[53] R. P. Dziak, J.-Y. Royer, J. H. Haxel, M. Delatre, and D. R. Bohnenstiehl et al., "Hydroacoustic detection of recent seafloor volcanic activity in the southern Indian Ocean," in *Transactions, American Geophysical Union, Fall Meeting, T13. San Francisco, CA (abstract)*, 2008, pp. 1–1.

[54] E. Tsang-Hin-Sun, J.-Y. Royer, and J. Perrot, "Seismicity and active accretion processes at the ultraslow-spreading southwest and intermediate-spreading southeast indian ridges from hydroacoustic data," *Geophysical Journal International*, vol. 206, no. 2, pp. 1232–1245, 2016.

[55] E. Leroy, F. Samaran, J. Bonnel J.-Y. Royer, "Identification of two potential whale calls in the southern Indian Ocean, and their geographic and seasonal occurrence," *J. Acoust. Soc. Am.*, vol. 142, no. 3, pp. 1413–1427, 2017.

[56] Rankin S., Ljungblad D., Clark C., and Kato H., "Vocalisations of antarctic blue whales, balaenoptera musculus intermedia, recorded during the 2001/2002 and 2002/2003 iwc/sower circumpolar cruises, area v, Antarctica," *Journal of Cetacean Research and Management*, vol. 7, pp.13–20, 2005.

[57] "Sercel," http://www.sercel.com/, Accessed: 2017-03-27.

TABLES

Class	Training sig.	Test sig.	Total
Z-call	100	154	254
Mad1	100	164	264
Mad2	100	187	287
20Hz-pulse	100	900	1000
D-call	100	280	380
Noise	-	15000	15000

Table I. Number of training and test signals used for each class and for each iteration of the cross-validation.

	Z-call	Mad1	Mad2	20Hz-pulse	D-call
Z-call	**100**	0.00	0.00	0.00	0.00
	0.10	0.00	0.10	0.00	0.10
Mad1	0.00	**97.7**	1.90	0.00	0.40
	0.00	1.10	1.10	0.00	0.50
Mad2	0.00	0.30	**99.60**	0.00	0.10
	0.00	0.30	0.30	0.10	0.20
20Hz-pulse	0.00	0.00	0.00	**100**	0.00
	0.00	0.00	0.00	0.00	0.00
D-call	0.00	0.00	0.00	0.00	**100**
	0.00	0.10	0.10	0.00	0.10

Table II. Confusion matrix of the SINR-SRC algorithm (in %) without the rejection option. For each class, the upper line contains the mean and the lower line the standard deviation obtained for 100 cross-validation trials on the call library only.

	Z-call	Mad1	Mad2	20Hz-Pulse	D-call
Z-call	**79.89**	0.00	19.66	0.45	0.00
	15.96	0.00	16.06	0.44	0.00
Mad1	0.25	**96.77**	2.70	0.00	0.29
	0.66	1.44	1.22	0.00	0.33
Mad2	3.42	0.69	**95.89**	0.00	0.00
	3.09	0.37	3.14	0.00	0.00
20Hz-Pulse	0.01	0.00	0.00	**93.00**	6.99
	0.02	0.00	0.02	5.13	5.14
D-call	3.73	0.00	0.00	0.00	**96.27**
	1.17	0.00	0.00	0.00	1.17

Table III. Confusion matrix (in %) for the method derived from [29] without rejection option. For each class, the upper line contains the mean and the lower line the standard deviation obtained for 100 cross-validation trials on the call library only.

	Z-call	Mad1	Mad2	20Hz-pulse	D-call	Rejected
Z-call	**99.62**	0.00	0.00	0.00	0.00	0.38
	0.44	0.00	0.00	0.00	0.00	0.44
Mad1	0.00	**96.92**	0.52	0.00	0.00	2.56
	0.00	1.27	0.56	0.00	0.00	1.07
Mad2	0.00	0.35	**98.11**	0.00	0.00	1.54
	0.00	0.28	0.73	0.00	0.00	0.64
20Hz-Pulse	0.00	0.00	0.01	**97.63**	0.00	2.36
	0.00	0.00	0.03	0.72	0.00	0.72
D-call	0.00	0.01	0.00	0.00	**89.89**	10.1
	0.00	0.04	0.00	0.00	1.81	1.81
Noise	0.75	0.79	3.21	0.27	1.64	**93.34**
	0.39	0.62	1.96	0.17	1.89	4.65

Table IV. Confusion matrix of the SINR-SRC algorithm (in %) with the rejection option activated. The false alarm probability specified on the SINR distributions after injection of filtered Gaussian noise samples into the dictionaries is 1%. For each class, the upper line contains the mean and the lower line the standard deviation obtained for 100 cross-validation trials on the call and noise library.

	Z-call	Mad1	Mad2	20Hz-pulse	D-call
Noise	11.76	4.97	35.08	21.70	26.49
	19.61	7.34	27.92	29.49	30.89

Table V. Classification results of SINR-SRC (in %) with noise inputs only. The rejection option is deactivated. The upper line contains the mean and the lower line the standard deviation obtained for 100 cross-validation trials.

	Z-call	Mad1	Mad2	20Hz-Pulse	D-call	Rejected
Z-call	**75.64**	0.00	0.01	0.00	0.00	24.35
	14.91	0.00	0.06	0.00	0.00	14.91
Mad1	0.01	**87.98**	1.13	0.00	0.00	10.88
	0.09	3.70	0.61	0.00	0.00	3.55
Mad2	1.93	0.43	**90.60**	0.00	0.00	7.04
	1.87	0.32	3.88	0.00	0.00	3.26
20Hz-Pulse	0.01	0.00	0.00	**85.08**	0.00	14.92
	0.02	0.00	0.00	5.30	0.00	5.30
D-call	3.73	0.00	0.00	0.00	**88.26**	8.01
	1.17	0.00	0.00	0.00	2.51	2.32
Noise	4.94	0.00	21.60	0.00	7.05	**66.41**
	5.13	0.00	16.68	0.00	5.21	24.62

Table VI. Confusion matrix (in %) for the method derived from [29] with the rejection option activated. The rejection threshold is 3 on the Mahalanobis distance between feature vectors and assigned mean attributes. For each class, the upper line contains the mean and the lower line the standard deviation obtained for 100 cross-validation trials on the call and noise library.

LIST OF FIGURES

LIST OF TABLES