Applied Acoustics 243 (2026) 111103

Contents lists available at ScienceDirect

Applied Acoustics

journal homepage: www.elsevier.com/locate/apacoust

The computation of generalized embeddings for underwater acoustic target recognition using contrastive learning

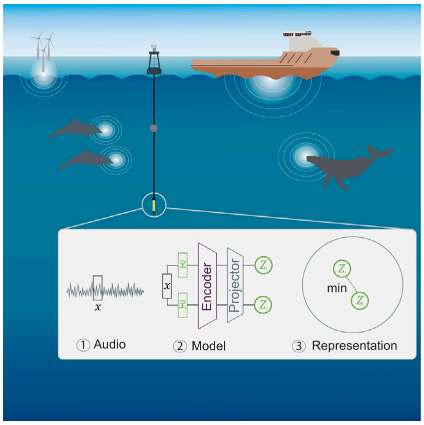
Hilde I. Hummel[a,*], Arwin Gansekoele[a], Sandjai Bhulai[b], Rob van der Mei[a,b]

[a] Centre of Mathematics and Computer Science, Department of Stochastics, Science Park 123, Amsterdam, 1098 XG, Netherlands
[b] Vrije Universiteit, Department Mathematics, De Boelelaan 1111, Amsterdam, 1081 HV, Netherlands

HIGHLIGHTS

- An application of unsupervised contrastive learning (CL) on unlabeled underwater acoustic dataset for Underwater Acoustic Target Recognition. It demonstrates a novel implementation on abundantly available unlabeled data collected from a single hydrophone for the first time.
- This novel framework effectively translates unstructured acoustic data into generalized embeddings which can be used for various downstream classification tasks.
- The proposed unsupervised CL framework produces robust and generalized embeddings. This creates the potential for this framework to translate to other underwater acoustic analysis tasks, such as climate change monitoring and nuclear bomb test detection.

GRAPHICAL ABSTRACT



ARTICLE INFO

2000 MSC:
0000
1111

PACS:
0000
1111

Keywords:
Ship radiated noise
Unsupervised learning
Contrastive learning
Underwater sound
Marine mammals
UATR

ABSTRACT

The increasing level of sound pollution in marine environments poses an increased threat to ocean health, making it crucial to monitor underwater noise. By monitoring this noise, the sources responsible for this pollution can be mapped. Monitoring is performed by passively listening to these sounds. This generates a large amount of data records, capturing a mix of sound sources such as ship activities and marine mammal vocalizations. Although machine learning offers a promising solution for automatic sound classification, current state-of-the-art methods implement supervised learning. This requires a large amount of high-quality labeled data that is not publicly available. In contrast, a massive amount of lower-quality unlabeled data is publicly available, offering the opportunity to explore unsupervised learning techniques. This research explores this possibility by implementing an unsupervised Contrastive Learning approach. Here, a Conformer-based encoder is optimized by the so-called *Variance-Invariance-Covariance Regularization* loss function on these lower-quality unlabeled data and the translation to the labeled data is made. Through classification tasks involving recognizing ship types and marine mammal vocalizations, our method demonstrates the ability to produce robust and generalized embeddings. This shows the potential of unsupervised methods for various automatic underwater acoustic analysis tasks.

[*] Corresponding author.
Email address: h.i.hummel@cwi.nl (H.I. Hummel).

https://doi.org/10.1016/j.apacoust.2025.111103
Received 16 May 2025; Received in revised form 29 August 2025; Accepted 24 September 2025

Available online 15 October 2025

1. Introduction

The ocean environment is polluted by the increased level of artificially induced sounds. This negatively impacts the ocean health directly. For this reason, monitoring underwater acoustics is crucial to protect our ocean environments. By monitoring the underwater sound, the sources of noise responsible for pollution can be mapped. One such system to monitor ocean sound is the implementation of a passive acoustic monitoring system. Within such a system, hydrophones are placed to passively listen to the underwater acoustics. These acoustics originate from multiple sources, such as ships, marine mammals, surface waves, and rain. Due to the mix of acoustic sources and complexity of the acoustic ocean environment, the analysis of underwater recordings becomes challenging. In addition, these systems generate a large amount of data, creating the need for automatic classification of underwater acoustic sounds, called underwater acoustic target recognition (UATR). To automate UATR, machine learning has shown its potential [1–4].

Until now, the development of UATR algorithms has focused on supervised learning, which requires a vast amount of high-quality labeled acoustic data for training. Unfortunately, the amount of publicly available labeled acoustic data is limited. In contrast, a large amount of unlabeled underwater acoustic data is available to the public. Until now, no research has been conducted on the application of these lower-quality data utilizing unsupervised learning models for UATR. One promising unsupervised approach to exploit these unlabeled acoustic data is Contrastive Learning (CL). This method tries to maximize the similarity between positive data samples, while minimizing the similarity for unrelated data samples. Within the computer vision domain, several CL methods are proposed to achieve remarkable performance using unlabeled datasets, such as SimCLR [5] and Momentum Contrast (MoCo) [6].

In addition to computer vision tasks, this specific method has excellent performance in general audio tasks, as seen in [7–9]. Furthermore, multiple studies have already implemented a form of CL to recognize the unique acoustic signatures of ships. For example, [10] proposed a ResNet-based encoder and optimized it in a supervised setting. In this study, they forced the recordings of the same ship to be close together in the embedding space while pushing the recordings of other vessels further away. They built a distance-based classifier on top of their CL framework to avoid overfitting under limited data. Similarly, [11] proposed a three-layer *Multi-Layer Perceptron* (MLP) optimized using the supervised contrastive loss function [12]. They did not include any augmentations and only used the label information, achieving over 98 % accuracy on ship type classification. Another supervised CL framework for ship type classification was proposed in [13]. In this study, they proposed two separate ResNet encoders, where both encoders process two different transformations of the same audio. They implemented a combined loss of cosine similarity and the cross-entropy loss, achieving 80 % accuracy. In addition to these supervised methods, [14] proposed a few-shot learning method to cope with limited labeled data. In this study, they proposed a four-stage training scheme inspired by SimCLR [5] and BYOL [15]. They treated a dataset of labeled ship recordings as partially unlabeled and explored the impact of reducing the number of labeled data samples on their proposed method. In this study, they showed that they only needed 10 % of the labels to achieve the desired performance. A similar few-shot learning method was proposed in [16] using a MoCo-based framework to achieve over 96 % accuracy.

Although prior UATR studies report high recognition accuracy scores, their ability to generalize to similar tasks remains unexplored [17]. Generalization of models is needed to transfer the knowledge captured within the model to related tasks, making the generalized model applicable to various UATR tasks. It is an important property to enable more broadly applicable UATR. This study aims to address this limitation by generating generalized embeddings from publicly available unlabeled data. These embeddings are optimized by the implementation of CL. The ability of the model to generalize is tested by implementing a simple classifier on Deepship [18], ShipsEar [19] for ship type classification, and The Best of Watkin’s [20] for marine mammal sound classification. The contribution of this paper is three-fold:

- The first implementation of unsupervised CL on a separate unlabeled underwater acoustic dataset is presented.
- A translation from this raw unlabeled data from a single hydrophone to a labeled classification task is made.
- The unsupervised CL approach generates robust and generalized embeddings.

Altogether, this work demonstrates the potential of an unsupervised CL framework for automatic UATR trained on widely available unlabeled data. This eliminates the dependency on a vast amount of high-quality labeled data and can be applied to various UATR tasks. Furthermore, this proof-of-concept can be extended by including more freely available unlabeled underwater acoustic data and by introducing additional underwater acoustic analysis tasks.

2. Material and methods

A CL pipeline is proposed to generate generalized and informative embeddings, where the proposed architecture is visualized in Fig. 1. In this architecture, an encoder is defined to compute the embeddings, and an expander is added to project these embeddings to a higher-dimensional space for optimization. After training this pipeline, the generated embeddings are used to fit simple task-specific classification models.

2.1. Data

In this work, both labeled and unlabeled data are utilized to train and evaluate the model.

2.1.1. Unlabeled datasets

For the model’s training, a single hydrophone from the Ocean Network Canada (ONC) dataset [21] was selected. The hydrophone [1] is located in the bay near Vancouver and is characterized by abundant shipping activity (longitude -123.43 West and latitude 49.04 North at 298 m depth). This specific hydrophone was selected to align with the same site as the Deepship benchmark dataset. Therefore, the characteristics of the dataset are similar to those of the Deepship data. From the recordings of this selected hydrophone, multiple 5-minute recordings were selected within the time range of September 2019 to December 2020. To ensure comprehensive representation, the selection procedure ensured equal representation of every month and time of day (morning, afternoon, evening, and night), resulting in 30 GB of raw data. Together,

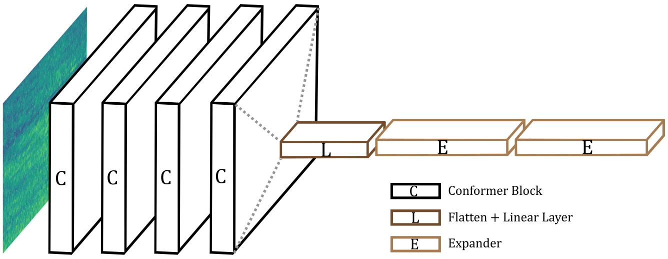


Fig. 1. The proposed architecture for the unsupervised framework. The framework starts with a Mel Spectrogram, followed by four Conformer Blocks (indicated in black), one flatten layer combined with a linear layer (indicated in brown), and finalized with an Expander (indicated in light brown).

[1] https://data.oceannetworks.ca/DataSearch?locationCode=SCVIP&deviceCategoryCode=HYDROPHONE

no information on the content of these data, like noise types or coverage of marine mammal sounds or ship types, is available, making the implementation of CL challenging (see Section 2.5).

2.1.2. Labeled datasets

For the evaluation of the proposed method, three labeled datasets were selected as benchmark. These datasets are widely used in related research, making it easier to compare the results with previous studies [2]. Each benchmark offers a different geographical area and labels, summarized as follows:

1. **Deepship** [18]: This benchmark dataset contains 47 hours and 3 min of single-ship recordings categorized into four classes. The recordings were made in the strait of Georgia near Vancouver.
2. **ShipsEar** [19]: A smaller dataset recorded near Port Ria de Vigo, totaling 3 hours and 8 min of single-ship recordings categorized into five classes.
3. **The Best of Watkin's** [20]: A marine mammal vocalization dataset categorized into 45 classes. All recordings were made over a span of seven decades in a wide range of oceanic areas.

The Deepship dataset was split into a training and test set using a time-wise split, ensuring that the training set represents the past and the test set represents the future. The recordings until November 2017 were used as training data and the recordings from December 2017 onward as test set. A time-wise split ensures that the proposed framework is most representative of practical systems, making a better evaluation for real-world deployment. However, due to this split, almost 60 % of the data samples in the test set are newly seen ships. This requires the model to learn generalizable embeddings to correctly classify these previously unseen ships. This property is crucial for generalized models, as they need to be applicable to various downstream tasks and therefore robust to distribution shifts [22]. The remaining labeled datasets had no available time information and therefore were divided into 80 % training data and 20 % test data.

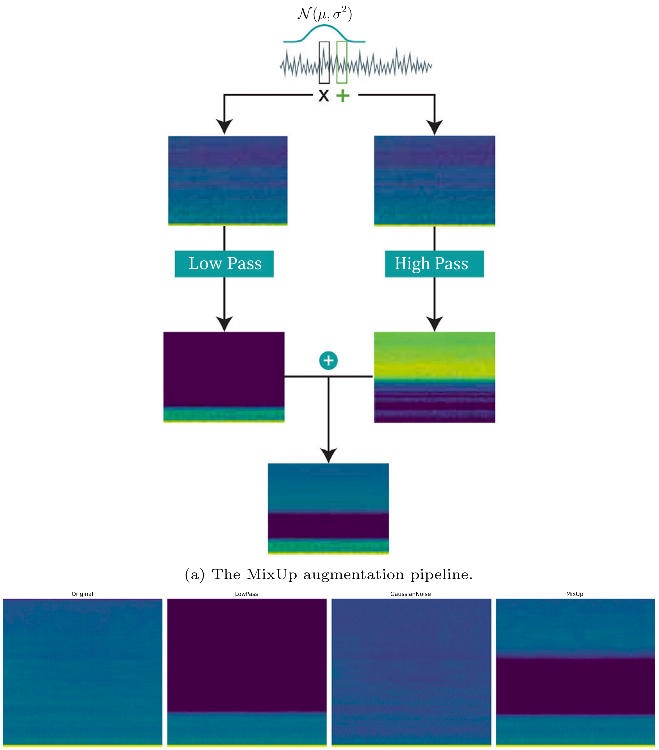
2.1.3. Preprocessing

After the datasets were split, the audio samples were downsampled to 16 kHz. Doing so reduces the dimensionality of the data while minimizing the loss of important information captured within the audio. From these downsampled audio, nonoverlapping two-second windows were created.

2.2. Augmentations

In a CL framework, it is necessary to define positive samples for training. Since no information on the unlabeled training data is available, the audio was augmented to generate positive samples. Here, these positive samples are defined by applying two individual augmentation functions to a single audio fragment in the time domain. These two augmentation functions are randomly chosen from a family of augmentation functions. This family consists of four different augmentation functions, namely:

1. **No augmentation:** Retains the original audio.
2. **Gaussian noise:** Adds random Gaussian noise and constrains the output SNR between 0.3 dB and 0.5 dB. The addition of Gaussian noise simulates noisier and diverse underwater acoustic environments.
3. **Low-pass filter:** Filters the input audio by a maximum of 1 kHz as the cutoff frequency. As stated in [2], the most discriminative information for the classification of the type of ships lies between 50 Hz and 1 kHz. Therefore, a low-pass filter is a representative augmentation function.
4. **Mixup strategy:** Combines two filtered audio samples and is inspired by the work of Ref. [23] who presented *Temporal Neighborhood Coding* for positive sample definition in time series data. The original audio sample is filtered by the 1 kHz low-pass filter as previously described. These lower frequency bands capture the most unique acoustic ship characteristics [2]. A second audio sample is selected by normal distribution over time, using the center time of the recording as μ and 50 s as σ. From this distribution, another two-second audio sample is drawn. This drawn sample is then filtered by a high-pass filter from 1 kHz-8 kHz, capturing related but different high-frequency band characteristics. Finally, these filtered samples are added, forming a mixed audio sample. A MixUp strategy is commonly applied in audio pipelines [24,25], this adds more contrast to lower-contrast spectrograms. The complete mixup pipeline is illustrated in Fig. 2a.



(a) The MixUp augmentation pipeline.

(b) Visualization for the proposed augmentation functions in the augmentation family.

Fig. 2. Visualization of the augmentation family proposed for defining positive samples during training. The top image shows the pipeline for the MixUp strategy and the down image shows the resulting spectrograms for all augmentation functions.

The proposed augmentation functions are visualized in Fig. 2b.

2.3. Conversion to mel spectrogram

The augmented audio is converted to a Mel spectrogram. This is a time-frequency representation of the time-domain audio convolved with the Mel scale. This Mel scale comprises multiple half-overlapping triangular filters designed to mimic the human perception of sound; in this study the number of Mel filters was set to 128. Each triangular filter is centered at frequency f_{mel} which is defined as:

$$f_{mel}(Hz) = 2595 \times \log_{10}\left(1 + \frac{f(Hz)}{700}\right). \quad (1)$$

2.4. Architecture

The architecture of the proposed method combines an encoder with an expander (see Fig. 1). In this research, a small architecture is chosen to create a proof of concept. In this regard, the encoder is defined by four Conformer blocks [26], each including feedforward layers, four multi-head self-attention heads, and convolutional modules with a kernel size

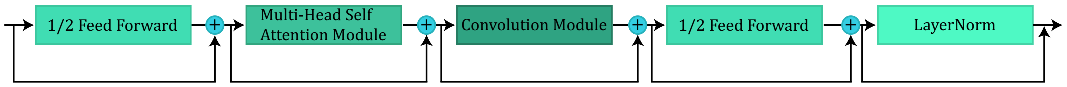


Fig. 3. A single Conformer Block.

of 31 (see Fig. 3). Previous research has demonstrated the potential of the Conformer model in audio classification on AudioSet [27]. The resulting embeddings are flattened to a 2048-size vector and passed through a linear layer. The architecture is finalized by the expander, composed of a two-layer MLP of dimension 8196.

2.5. Loss function

The unlabeled training dataset can be viewed as a single long recording. Since there is no information on the content of this recording available, it is difficult to assign negative samples as proposed in the original SimCLR paper [5]. For this reason, the *Variance-Invariance Covariance Regularization* (VICReg)[28] loss function is implemented. Here, no prior definition of negative samples is needed. The loss function consists of three components: the invariance, variance, and covariance components. A weighted average is taken from these components to define the loss function. The first component is the variance component ($v(Z)$), where Z is the batch:

$$v(Z) = \frac{1}{d}\sum_{j=1}^{d} abs\big(\gamma - S(z^j, \epsilon)\big), \tag{2}$$

where d is the number of features and z^j is the specified feature value of all batch samples. γ is a constant target value for the standard deviation, which is fixed at 1 as in the original paper. Finally $S(z^j, \epsilon)$ is given by:

$$S(x, \epsilon) = \sqrt{Var(x) + \epsilon}, \tag{3}$$

where ϵ is a small constant fixed to 0.0001. This function encourages the variance to be the same for all j in the current batch. The covariance matrix of Z is defined as:

$$C(Z) = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \bar{z})(z_i - \bar{z})^T, \tag{4}$$

where

$$\bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i. \tag{5}$$

The regularization of the covariance can then be defined as:

$$c(Z) = \frac{1}{d}\sum_{i\neq j}[C(Z)]^2_{i,j}. \tag{6}$$

This term encourages the off-diagonal components to be close to 0. This decorrelates the variable of each embedding, which prevents informational collapse. The invariance component is given by the negative cosine similarity:

$$s(Z_i, Z_i') = -\frac{Z_i \cdot Z_i'}{\|Z_i\|\|Z_i'\|}. \tag{7}$$

$$l(Z, Z') = \lambda s(Z, Z') + \mu[v(Z) + v(Z')] + v[c(Z) + c(Z')]. \tag{8}$$

In this loss function, the scaling components are hyperparameters that need to be tuned. The overall objective function is given by a summation over batches I in dataset D and over augmentations functions t and t' derived from the augmentation family T:

$$L = \sum_{I\in D}\sum_{t,t' \; T} l(Z^I, Z'^I). \tag{9}$$

2.6. Parameter selection

The proposed model is optimized for 30 epochs using a batch size of 2048. Due to this large batch size, the number of update steps for individual weights within a single epoch is reduced. This may be problematic if the number of epochs is not increased when the model is optimized by the commonly applied Adam optimizer. For this reason, the model was optimized using the Layerwise Adaptive Rate Scaling (LARS) optimizer[29] with a learning rate of 0.01. This optimizer updates layer-wise instead of the individual weights, reducing the need for more epochs during training. A decay was implemented, where the learning rate is reduced by a factor of 10 if learning stagnates. The weights of the loss function were set to $\lambda = 5$, $\mu = 5$, and $v = 1$. The complete pipeline was built using the PyTorch module and available on GitHub[2]. For classification, a logistic regression was used, using the learned embeddings as features. Finally, the performance of this classifier was evaluated using the accuracy score.

2.7. Baselines

The proposed unsupervised method was compared to a supervised CL method inspired by the SimCLR framework [5] in terms of accuracy and a F1 score weighted by the support of the class. This baseline consists of a ResNet18 encoder followed by a two-layer MLP of dimension 128 as the projector. A smaller baseline architecture is chosen to cope with the limited quantity of labeled data.

The baseline model is trained using the supervised contrastive loss function [12] This loss function is derived from the original NTXent loss [5] which takes the following form:

$$\mathcal{L}^{self} = \sum_{i\in I}\mathcal{L}_i^{self} = -\sum_{i\in I}\log\frac{\exp(\mathbf{Z_i}\cdot\mathbf{Z_j}/\tau)}{\sum_{a\in\mathcal{I}\setminus\{i\}}\exp(\mathbf{Z_i}\cdot\mathbf{Z_a}/\tau)} \tag{10}$$

in this function, let $i \in \mathcal{I} \equiv \{1..2\ N\}$ be the index of the sample of interest called the anchor sample and j be its augmented version and defined as the positive sample. All the other $2(N-1)$ samples are defined as the negatives and the temperature scalar parameter is defined as $\tau \in \mathcal{R}^+$.

This original function is rewritten to incorporate label information into the following format:

$$\mathcal{L}_{sup} = \sum_{i\in I}\frac{-1}{|\mathcal{P}(i)|}\sum_{p\in\mathcal{P}(i)}\log\frac{\exp(\mathbf{Z_i}\cdot\mathbf{Z_p}/\tau)}{\sum_{a\in\mathcal{I}\setminus\{i\}}\exp(\mathbf{Z_i}\cdot\mathbf{Z_a}/\tau)} \tag{11}$$

here, $\mathcal{P}(i)$ is the set of indices for data samples with the same label as anchor i. The loss is summed over all anchor samples within the batch and normalized by $\mathcal{P}(i)$ which accounts for the number of positive pairs for each anchor. This function aims to place the samples with the same label close together in the embedding space and pushes samples with other labels further away. The supervised baseline model is trained on the labeled Deepship data and is optimized using either the LARS optimizer with a learning rate of 0.01 or the Adam optimizer with a learning rate of 0.0005. The Adam optimizer is also implemented since this optimizer is suggested in previous work that proposed a supervised CL framework for UATR [10,11,13]

3. Results

In this section, the generalizability of the proposed unsupervised method is evaluated and compared to the supervised CL framework.

[2] https://github.com/hildeingvildhummel/UATR

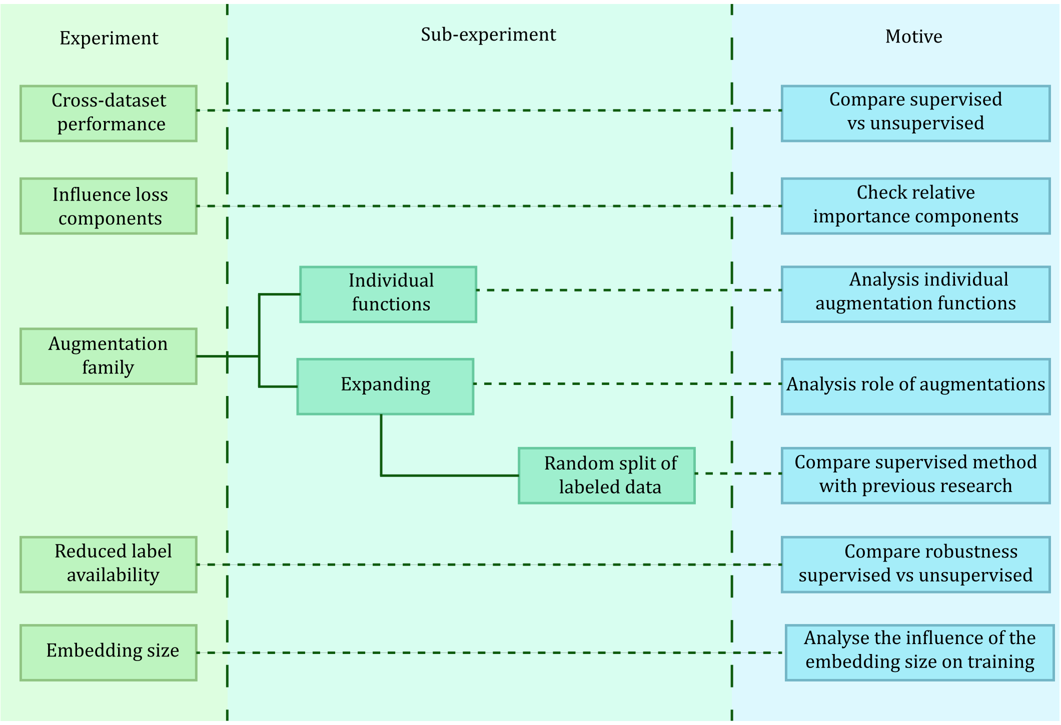

Experiment
Sub-experiment
Motive
Cross-dataset performance
Compare supervised vs unsupervised
Influence loss components
Check relative importance components
Individual functions
Analysis individual augmentation functions
Augmentation family
Expanding
Analysis role of augmentations
Random split of labeled data
Compare supervised method with previous research
Reduced label availability
Compare robustness supervised vs unsupervised
Embedding size
Analyse the influence of the embedding size on training

Fig. 4. Overview of the experiments.

Table 1
The accuracy for multiple architectures, methods, and optimizers reported over multiple benchmark datasets.

Dataset	Deepship	ShipsEar	Watkin's
Supervised CL ResNet18 (Adam)	53.94 %	21.40 %	54.54 %
Supervised CL ResNet18 (LARS)	**56.51 %**	43.94 %	80.46 %
Supervised Conformer (LARS)	55.16 %	48.47 %	**87.10 %**
Unsupervised ResNet18 (LARS)	50.34 %	42.25 %	71.80 %
Unsupervised Conformer (LARS)	54.87 %	**57.42 %**	86.10 %

Table 2
The weighted F1 score for multiple architectures, methods, and optimizers reported over multiple benchmark datasets.

Dataset	Deepship	ShipsEar	Watkin's
Supervised CL ResNet18 (Adam)	0.49	0.08	0.54
Supervised CL ResNet18 (LARS)	**0.55**	0.42	0.77
Supervised Conformer (LARS)	0.54	0.49	**0.85**
Unsupervised ResNet18 (LARS)	0.49	0.37	0.63
Unsupervised Conformer (LARS)	0.54	**0.57**	0.84

One way to estimate this is to measure cross-dataset performance. In this study, the datasets cover real-world recordings from various regions of the world with different ocean environments. In addition, an in-depth analysis of the weights of the VICReg loss components, augmentation functions, availability of labeled data, and embedding dimensions is presented. An overview of the experiments is visualized in Fig. 4.

3.1. Multi-purpose classification

First, performance was compared between supervised contrastive loss baseline models and unsupervised VICReg models. The supervised and unsupervised approaches were trained with a ResNet18 model and a Conformer model. The accuracy and weighted F1-score for the logistic regression models built on top of the backbones are given in Table 1 and Table 2. Note that the ResNet18 backbone model trained with supervised CL was optimized with both the Adam optimizer and the LARS optimizer. As mentioned previously, the Adam optimizer is still commonly used when building supervised CL frameworks for UATR. First, the results indicate that the use of a LARS optimizer improves the downstream performance of Deepship, which is also the dataset both backbones were trained on. Inclusion of the LARS optimizer also improves the generalizability of these backbone models to ShipsEar and Watkin's. In light of these results, corroborated by the fact that LARS-based optimizers are the standard when working with very large batch sizes, all further backbones were optimized using the LARS optimizer.

Further comparison is made between the ResNet18 model and the Conformer model. As the Conformer model has more parameters and is built specifically for audio, it is expected to perform better than the ResNet18 model. Although this hypothesis holds for the unsupervised approach, one surprising result is shown for the supervised CL ResNet18 on Deepship. Here, the ResNet18 supervised CL approach achieves the best performance in Deepship, even compared to the Conformer supervised CL approach. However, the supervised CL Conformer translates better to other datasets. While the ResNet18 supervised CL model is more capable of capturing the properties important for Deepship classification, the Conformer has learned a more general backbone model.

Finally, the difference between using a form of supervised CL and a form of unsupervised CL is compared. For ResNet18, using supervised CL on Deepship builds both a better backbone for Deepship and a model more generalizable to other datasets. However, the trade-off becomes a bit more nuanced when comparing the Conformer models. The supervised CL Conformer model is clearly more capable on Deepship. This result is not surprising, as its backbone was trained on Deepship and thus already optimized to discriminate Deepship classes. On the other hand, the unsupervised Conformer achieves a similar performance on Watkin's and a better performance on ShipsEar. The features learned by the unsupervised CL approach thus seem more generalizable to other datasets, a t-SNE plot of the embedding space using Deepship is visualized in Fig. 5.

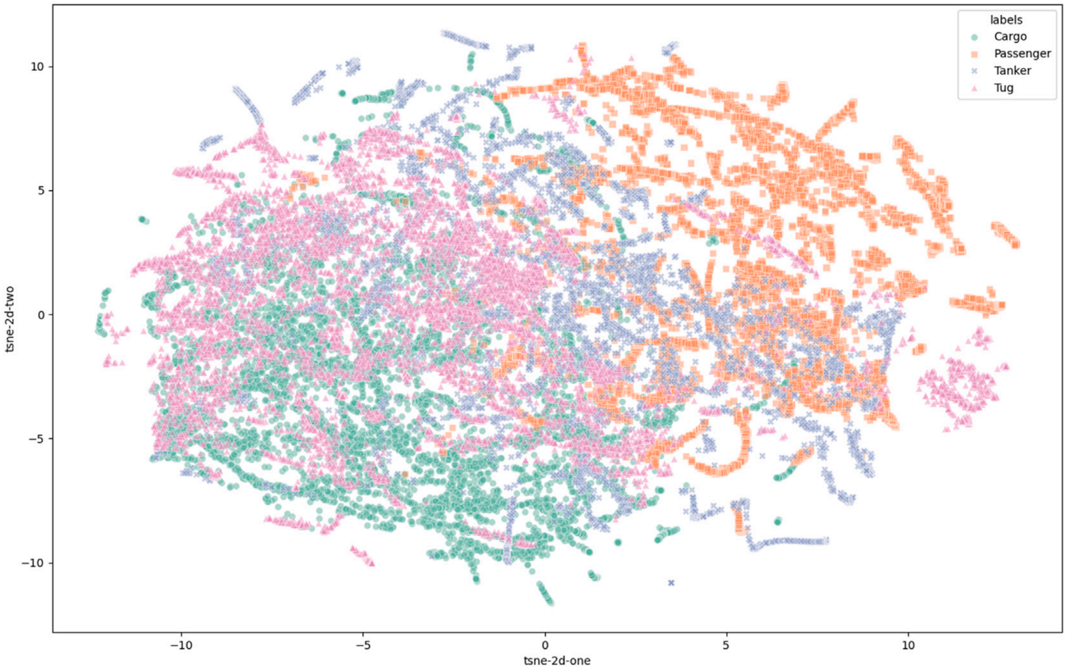


Fig. 5. t-SNE of Deepship test set embeddings generated by the unsupervised Conformer model.

When the model capacity becomes larger, the use of a dataset such as Deepship to train a backbone model seems to bias it towards that dataset.

3.2. Influence individual loss components of VICReg

In the original VICReg paper [28], the authors demonstrated that at least the invariance and the variance components are essential to prevent collapse during training. They originally recommended a weight combination of $\lambda = 25$, $\mu = 25$, and $\upsilon = 1$, focusing on computer vision tasks. However, images encounter more contrast than spectrograms, making UATR more susceptible to *informational collapse* in which the variables of the embeddings carry redundant information [28]. Consequently, the covariance component of the VICReg loss function can be more important for underwater acoustic data than for computer vision-related tasks. For this reason, various recommended weight combinations were tested from the appendix of the work by Ref. [28] (see Table 3). This confirmed that the relative importance of the covariance component needs to be increased to optimize the embeddings in UATR.

3.3. Influence of the augmentation family

An augmentation family, consisting of various augmentation functions, is needed to define positive samples for unsupervised CL training. During training, a single audio recording is augmented by randomly selecting two individual functions from this family. Then both augmented samples will be optimized to be close together in the expanded space using CL. The performance of a model optimized using CL, may be sensitive to the choice of the augmentation functions. For this reason, the influence of the different augmentation functions within the specified family is analyzed (see Tables 4 and 5). The highest accuracy scores on both Deepship and Watkin's are achieved by the LowPass filter. This augmentation function alone achieves performance similar to that of the model trained utilizing the full augmentation family. This is likely due to the fact that the most discriminative information in the spectrogram is primarily located in the lower frequency bands. This result indicates that the unsupervised model focuses on the most informative spectral regions. However, the performance of the model solely using the LowPass filter is reduced in ShipsEar. This suggests that the combination of all the augmentation functions is needed to achieve stable cross-dataset performance.

Table 3
The influence of the weights assigned to the individual loss components of VICReg on the classification of Ship types.

λ	μ	υ	Accuracy
1	1	1	55.35 %
5	5	1	**55.61 %**
25	25	1	23.81 %

Table 4
The influence of the augmentation functions on the accuracy for the classification of Ship types.

Dataset	Deepship	ShipsEar	Watkin's
LowPass	**55.93 %**	49.42 %	**87.00 %**
Noise	52.54 %	50.19 %	85.21 %
MixUp	55.25 %	51.86 %	86.03 %
Combined	55.61 %	**52.14 %**	**87.00 %**

Table 5
The influence of the augmentation functions on the weighted F1-score for the classification of Ship types.

Dataset	Deepship	ShipsEar	Watkin's
LowPass	0.54	0.44	0.83
Noise	0.52	0.48	0.83
MixUp	0.54	**0.51**	**0.84**
Combined	**0.55**	0.50	0.83

3.3.1. Expanding the augmentation family

The augmentation family was expanded by multiple speech-based augmentation functions, such as varying gain, reverb, pitch, and performing polarity inversion. However, introducing more types of speech-based augmentations does not improve the unsupervised model

Table 6
The accuracy of both the Supervised and Unsupervised frameworks with an expanded augmentation family, incorporating speech-based augmentation functions.

Dataset	Deepship	ShipsEar	Watkin's
Supervised CL ResNet18 (Adam)	**63.07 %**	45.91 %	75.73 %
Supervised CL ResNet18 (LARS)	61.17 %	**52.64 %**	86.33 %
Supervised CL Conformer (LARS)	56.55 %	45.41 %	**86.94 %**
Unsupervised CL ResNet 18 (LARS)	42.48 %	38.24 %	69.66 %
Unsupervised CL Conformer (LARS)	57.78 %	50.47 %	86.14 %

Table 7
The weighted F1-score of both the Supervised and Unsupervised frameworks with an expanded augmentation family, incorporating speech-based augmentation functions.

Dataset	Deepship	ShipsEar	Watkin's
Supervised CL ResNet18 (Adam)	**0.63**	0.43	0.71
Supervised CL ResNet18 (LARS)	0.60	**0.49**	0.84
Supervised CL Conformer (LARS)	0.56	0.44	**0.85**
Unsupervised CL ResNet 18 (LARS)	0.39	0.33	0.62
Unsupervised CL Conformer (LARS)	0.56	0.48	0.84

performance on all three datasets (see Tables 6 and 7). The performance of the unsupervised conformer on Deepship is improved; however, the performance on the other datasets is decreased. This shows the importance of representative augmentation functions in guiding the model in an unsupervised learning framework. In contrast, in the supervised setting, these models benefited from the additional augmentation functions. Here, both models show an increase in performance on all three datasets. The performance is greatly improved, showing that the generalization capacity of the supervised models is increased by expanding the augmentation family. The supervised ResNet18 model trained with the LARS optimizer has a competitive performance similar to the unsupervised learning model with the original augmentation family. Expanding the augmentation family increases the variability of the data during training, since the supervised method can rely on label information, this method can benefit from this additional variability. This is not the case for the unsupervised method, which does not depend on high-quality labeled data.).

3.3.2. Random data split for the supervised model

So far, the models are evaluated using a time-wise split for training and test set. However, previous research on automatic ship recognition adopted a random split for evaluation [2]. Unfortunately, such a split provides a poor reflection of the performance of the model in real-world conditions. To examine the impact of the train-test split, a random split was generated by randomly splitting the individual Deepship recordings into a training set (80 %) and a test set (20 %), ensuring that no individual recordings were present in both sets. The baseline models were trained in a supervised manner using the expanded augmentation family. Next, the performance of the models is examined by translating them again to ShipsEar and Best of Watkin's (see Fig. 6). Here, both ResNet18 models have an increased accuracy score when trained on the Random split, with a comparable performance to previous research [2]. This is because the number of newly seen ships is smaller compared to the time-wise split. In addition, the impact of the different seasons is reduced with the random split. However, the generalization performance for both ShipsEar and Best of Watkin's is worse when the ResNet18 is optimized on the random split. An explanation for this could be that the Deepship data is more stationary compared to ShipsEar and Watkins, which capture more temporal fluctuations (see Fig. 7). On the other hand, the supervised conformer model achieves the lowest performance on Deepship for both the time-wise and the random split. This is probably due to the model size, the conformer model is significantly bigger compared to ResNet18 and therefore the Deepship dataset is probably too small to properly train the model. The performance of the model trained on the random split is slightly higher on ShipsEar and has a similar performance on Watkin's. This shows that the defined split in the literature results in a higher reported accuracy on Deepship, however it may reduce its generalization capacity in other related tasks or regions.

3.4. Impact of reduced labeled data

The performance of the supervised models is known to rely on a large amount of labeled data. To evaluate the impact of reducing labeled data on model performance, the size of the Deepship training dataset was reduced by randomly selecting a subset of individual recordings for training. The unlabeled data used for unsupervised CL remained the same. This experiment explores the relative impact of a reduced set of available labeled data for both methods. The impact on performance for both the baseline models and the unsupervised models are visualized in Fig. 8. Here, the unsupervised Conformer method outperforms the supervised methods when the number of labeled data samples is reduced. This observation highlights the robustness of the unsupervised method compared to supervised models. A notable exception occurs at the 10 % dataset size, where the ResNet18 model trained with the Adam optimizer and the supervised Conformer exhibit better performance than the unsupervised model. This result is likely due to the redundancy of individual ships in the 10 % dataset. Here, 50 % of the individual ships in the 10 % training dataset are also present in the test set. Due to limited training

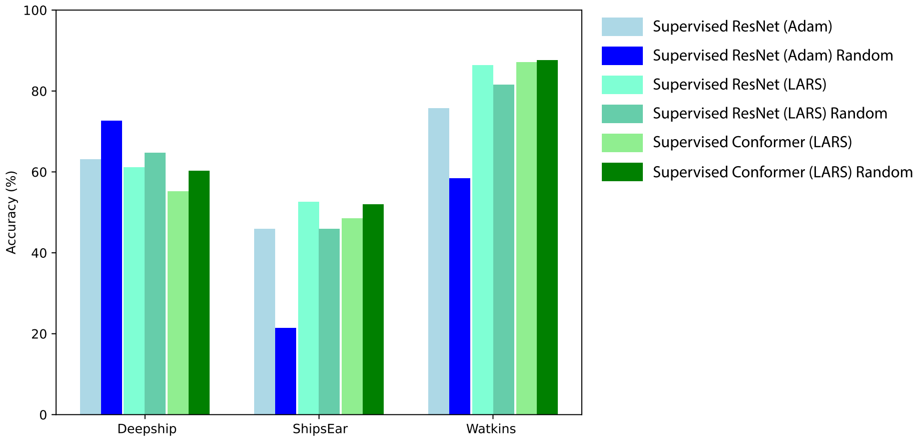


Fig. 6. Accuracy of supervised baseline models trained on either time-wise split or random split of Deepship.

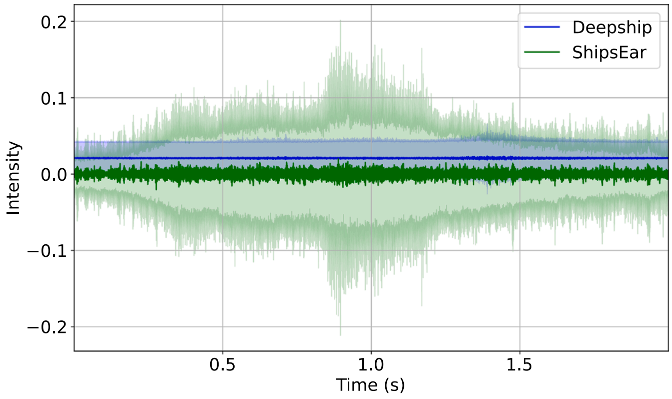


Fig. 7. Temporal Variation of Deepship and ShipsEar, showing the mean and the variability.

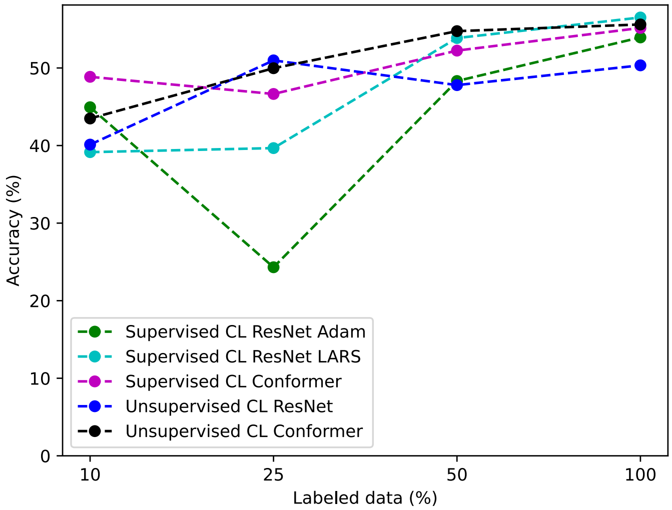


Fig. 8. The accuracy on Deepship when trained on reduced amount of labeled data.

data, the classifier can overfit to these samples, creating a performance peak for this configuration.

3.5. Influence of embedding size

The size of the embedding vector directly influences the ability of the model to learn the patterns within the dataset. Larger vectors have the ability to store more features; however, they increase the risk of underfitting the data and need more computational resources. However, smaller vectors may not capture essential patterns within the training dataset. This trade-off is examined in the proposed unsupervised framework, with the results illustrated in Fig. 9. Here, the accuracy score increases as the embedding size increases to 512, with only slight gains beyond this point. Further examination of the loss components during training showed that larger embedding sizes are associated with higher covariance loss and lower variance loss, while smaller sizes exhibit the opposite trend. These results align with the idea that larger vectors ease higher variations among them but also increase the likelihood of correlated elements, making the model prone to *informational collapse*.

3.6. Computational complexity

To evaluate the cost of running these models, the parameter count and the mean inference time were recorded for both (see Table 8). The mean inference time was based on the minimum time to process a sample calculated over 10 passes of the Deepship evaluation dataset. These numbers were recorded on a compute node consisting of 9 cores of an Intel Xeon GPU, 60GB of RAM memory, and an A100 40GB GPU. The training of all models took at most 10 hours under this setup.

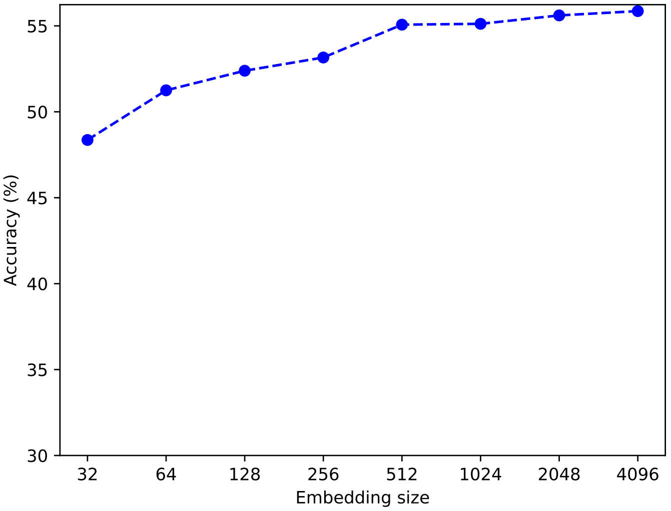


Fig. 9. The accuracy of the unsupervised conformer model on Deepship over the embedding size.

Table 8
Number of parameters and inference time.

	Parameter Count	Minimum Inference Time (ms)
ResNet18	16,690,544	2.817
Conformer	101,177,392	5.268

Note that the minimum inference time was taken per sample. This is the standard choice when recording the runtime of a task. The minimum time is closest to the true inference time, as variance is usually caused by external factors, such as other running processes. Here, we observe that processing a sample for both models only takes a few milliseconds. Although the Conformer model is around six times larger than the ResNet18 model, its inference time is less than double. This shows that scaling to larger models is viable due to the high degree of parallelizability. While the setup is quite large, the low inference times indicate that a real-time edge solution may be viable as well. Both models can process a sample of two seconds within a couple of milliseconds, demonstrating the computational viability of AI-based approaches for UATR.

4. Discussion

In this paper, a new proof-of-concept small architecture is proposed on unlabeled data. The model successfully extracted informative features from this unlabeled dataset and was able to transfer this knowledge to other labeled datasets by classifying both the sounds of marine mammals and the types of ships. In particular, the unsupervised model correctly classified newly seen ships into their corresponding class. The proposed unsupervised framework shows performances comparable to the supervised baselines. However, the performance of these supervised models is much more reliant on the amount of available labeled data. As the unsupervised framework does not require labeled data to extract informative features, its performance does not degrade as drastically when less labeled data is available.

The choice of augmentation functions proved important for the performance and generalizability of the backbone models. Using multiple augmentations tailored specifically to underwater acoustics resulted in a strong performance on all datasets for both supervised CL and unsupervised CL. The additional inclusion of speech-based augmentation functions gave mixed results with a marked improvement in accuracy for the smaller supervised CL models. Furthermore, this work used a time-wise split as the standard split for Deepship, as opposed to the commonly used random split. Using a time-wise split reduced the accuracy and F1-scores on Deepship. However, this work argues that a time-wise split provides a more realistic view of the real-world performance of these models.

The defined ship types in the labeled datasets have similar acoustic profiles with a high spread of acoustic signatures within a single class and a great overlap between classes. This makes the categorization of the acoustic profiles into these classes challenging and implies that a more suitable categorization of these profiles is needed. A solution to this problem could be to create classes based on the characteristics of the ship, such as the number of blades, the size of the ship, and the type of engine(s), instead of the first-level *Automatic Identification System* (AIS) categorization.

Overall, the proposed method achieves a performance similar to that presented in [14]. However, they reported a larger framework for the model, and they implemented few-shot learning using only Deepship data treating part of the data as unlabeled and not by making the translation between two different datasets. An advantage of this unsupervised approach over few-shot learning is that there is a great amount of unlabeled data available. Therefore, this framework can be extended by incorporating more diverse underwater acoustical data from multiple publicly available hydrophones. The results demonstrate that large datasets require models with more capacity to achieve sufficient generalization. In addition, the runtime analysis demonstrates that larger models do not necessarily pose issues when it comes to inference time, as processing a 2-second sample takes milliseconds. By incorporating more data, this framework could translate better to other underwater acoustic analysis tasks, such as climate change monitoring and nuclear bomb test detection.

5. Conclusion

This work highlights the potential of unsupervised CL methods in UATR. It explores the implementation of abundantly available unlabeled lower-quality underwater acoustic data and presents a CL pipeline for robust generalized embedding generation with competitive performance to supervised models. Overall, this shows the potential for the possibilities of unsupervised learning methods in the automatic analysis of large amounts of passive acoustic monitoring recordings. The proposed unsupervised CL has shown to be generalizable across multiple datasets and related underwater acoustic tasks. This makes the pipeline extendable for multiple automatic underwater acoustic analysis tasks. This creates the possibility to automatically identify and quantify sound pollution without depending on a large volume of high-quality labeled data.

CRediT authorship contribution statement

Hilde I. Hummel: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Conceptualization. **Arwin Gansekoele:** Writing – original draft, Conceptualization. **Sandjai Bhulai:** Writing – review & editing, Supervision, Conceptualization. **Rob van der Mei:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data link is shared in the manuscript

References

[1] Thomas M, Martin B, Kowarski K, Gaudet B, Matwin S. Marine mammal species classification using convolutional neural networks and a novel acoustic representation. In: Machine learning and knowledge discovery in databases: European conference, ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, proceedings, part III; Springer; 2020. p. 290–305.

[2] Hummel HI, van der Mei R, Bhulai S. A survey on machine learning in ship radiated noise. Ocean Engineering 2024;298:117252.

[3] Licciardi A, Carbone D. Whalenet: a novel deep learning architecture for marine mammals vocalizations on Watkins Marine Mammal Sound Database. IEEE Access 2024.

[4] Hamard Q, Pham M-T, Cazau D, Heerah K. A deep learning model for detecting and classifying multiple marine mammal species from passive acoustic data. Ecol Inform 2024:102906.

[5] Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: International conference on machine learning; PMLR; 2020. p. 1597–607.

[6] He K, Fan H, Wu Y, Xie S, Girshick R. Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2020. p. 9729–38.

[7] Saeed A, Grangier D, Zeghidour N. Contrastive learning of general-purpose audio representations. In: *Icassp* 2021 - 2021 *ieee* international conference on acoustics, speech and signal processing (*Icassp*); 2021. p. 3875–9. https://doi.org/10.1109/ICASSP39728.2021.9413528

[8] Fonseca E, Ortego D, McGuinness K, O'Connor NE, Serra X. Unsupervised contrastive learning of sound event representations. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); IEEE; 2021. p. 371–5.

[9] Niizumi D, Takeuchi D, Ohishi Y, Harada N, Kashino K. Byol for audio: exploring pre-trained general-purpose audio representations. IEEE ACM Trans Audio Speech Lang Process 2022;31:137–51.

[10] Nie L, Li C, Wang H, Wang J, Zhang Y, Yin F, Marzani F, Bozorg Grayeli A. A contrastive-learning-based method for the few-shot identification of ship-radiated noises. J Mar Sci Eng 2023;11(4):782.

[11] Sun B, Luo X. Underwater acoustic target recognition based on automatic feature and contrastive coding. IET Radar Sonar Navig 2023;17(8):1277–85.

[12] Khosla P, Teterwak P, Wang C, Sarna A, Tian Y, Isola P, Maschinot A, Liu C, Krishnan D. Supervised contrastive learning. Adv Neural Inf Process Syst 2020;33:18661–73.

[13] Xie Y, Ren J, Xu J. Guiding the underwater acoustic target recognition with interpretable contrastive learning. In: Oceans 2023-Limerick; IEEE; 2023. p. 1–6.

[14] Tian S, Bai D, Zhou J, Fu Y, Chen D. Few-shot learning for joint model in underwater acoustic target recognition. Scientific Reports 2023;13(1):17502.

[15] Grill J-B, Strub F, Altché F, Tallec C, Richemond P, Buchatskaya E, Doersch C, Avila Pires B, Guo Z, Gheshlaghi Azar M, et al. Bootstrap your own latent-a new approach to self-supervised learning. Adv Neural Inf Process Syst 2020;33:21271–84.

[16] Lin L, Chen Y, Wang F. Underwater passive target recognition based on self-supervised contrastive learning. In: 2024 3rd international conference on artificial intelligence, internet of things and cloud computing technology (AIoTC); IEEE; 2024. p. 375–80.

[17] Müller N, Reermann J, Meisen T. Navigating the depths: a comprehensive survey of deep learning for passive underwater acoustic target recognition. IEEE Access 2024.

[18] Irfan M, Jiangbin ZHENG, Ali S, Iqbal M, Masood Z, Hamid U. Deepship: an underwater acoustic benchmark dataset and a separable convolution based autoencoder for classification. Expert Syst Appl 2021;183:115270.

[19] Santos-Domínguez D, Torres-Guijarro S, Cardenal-López A, Pena-Gimenez A. Shipsear: an underwater vessel noise database. Appl Acoust 2016;113:64–9.

[20] Sayigh L, Daher MA, Allen J, Gordon H, Joyce K, Stuhlmann C, Tyack P. The Watkins Marine Mammal Sound Database: an online, freely accessible resource. In: Proceedings of meetings on Acoustics, vol. 27. AIP Publishing; 2016.

[21] Ocean Network Canada. I'm sorry, but it seems there might be some confusion in your request. Could you please provide the text you would like me to convert to sentence case?. 2007, https://data.oceannetworks.ca/home [accessed: 1 May 2023.

[22] Van Merriënboer B, Hamer J, Dumoulin V, Triantafillou E, Denton T. Birds, bats and beyond: evaluating generalization in bioacoustics models. Frontiers in Bird Science 2024;3:1369756.

[23] Tonekaboni S, Eytan D, Goldenberg A. Unsupervised representation learning for time series with temporal neighborhood coding, arXiv preprint 2021 arXiv:2106.00750.

[24] Kahl S, Wood CM, Eibl M, Klinck H. BirdNET: a deep learning solution for avian diversity monitoring. Ecol Inform 2021;61:101236.

[25] Xu Q, Jiang J, Xu K, Dou Y, Gao C, Zhu B, You K, Zhu Q. Self-supervised learning-for underwater acoustic signal classification with mixup. IEEE J Sel Top Appl Earth Obs Remote Sens 2023.

[26] Gulati A, Qin J, Chiu C-C, Parmar N, Zhang Y, Yu J, Han W, Wang S, Zhang Z, Wu Y, et al. Conformer: convolution-augmented transformer for speech recognition, arXiv preprint 2020 arXiv:2005.08100.

[27] Srivastava S, Wang Y, Tjandra A, Kumar A, Liu C, Singh K, Saraf Y. Conformer-based self-supervised learning for non-speech audio tasks. In: ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing (ICASSP); IEEE; 2022. p. 8862–6.

[28] Bardes A, Ponce J, LeCun Y. Vicreg: variance-invariance-covariance regularization for self-supervised learning, arXiv preprint 2021 arXiv:2105.04906.

[29] You Y, Gitman I, Ginsburg B. Large batch training of convolutional networks, arXiv preprint 2017 arXiv:1708.03888.