12-1-2020

Ensemble of convolutional neural networks to improve animal audio classification

Loris Nanni

Yandre M.G. Costa

Rafael L. Aguiar

Rafael B. Mangolin

Sheryl Brahnam
Missouri State University

See next page for additional authors

Recommended Citation

Nanni, Loris, Yandre MG Costa, Rafael L. Aguiar, Rafael B. Mangolin, Sheryl Brahnam, and Carlos N. Silla. "Ensemble of convolutional neural networks to improve animal audio classification." EURASIP Journal on Audio, Speech, and Music Processing 2020 (2020): 1-14.

Nanni *et al. EURASIP Journal on Audio, Speech, and Music Processing* (2020) 2020:8
https://doi.org/10.1186/s13636-020-00175-3

EURASIP Journal on Audio, Speech, and Music Processing

RESEARCH Open Access

Ensemble of convolutional neural networks to improve animal audio classification

Loris Nanni[1], Yandre M. G. Costa[2], Rafael L. Aguiar[3], Rafael B. Mangolin[2], Sheryl Brahnam[4] and Carlos N. Silla Jr.[3*]

Abstract

In this work, we present an ensemble for automated audio classification that fuses different types of features extracted from audio files. These features are evaluated, compared, and fused with the goal of producing better classification accuracy than other state-of-the-art approaches without ad hoc parameter optimization. We present an ensemble of classifiers that performs competitively on different types of animal audio datasets using the same set of classifiers and parameter settings. To produce this general-purpose ensemble, we ran a large number of experiments that fine-tuned pretrained convolutional neural networks (CNNs) for different audio classification tasks (bird, bat, and whale audio datasets). Six different CNNs were tested, compared, and combined. Moreover, a further CNN, trained from scratch, was tested and combined with the fine-tuned CNNs. To the best of our knowledge, this is the largest study on CNNs in animal audio classification. Our results show that several CNNs can be fine-tuned and fused for robust and generalizable audio classification. Finally, the ensemble of CNNs is combined with handcrafted texture descriptors obtained from spectrograms for further improvement of performance. The MATLAB code used in our experiments will be provided to other researchers for future comparisons at https://github.com/LorisNanni.

Keywords: Audio classification, Texture, Deep learning, Handcrafted features, Ensemble of classifiers, Pattern recognition

1 Introduction

Sound classification has been assessed as a pattern recognition task in different application domains for a long time. However, new advances have changed the typical way these classifier systems can be organized. One pivotal milestone has been the popularization of graphics processing units (GPUs), devices that have made it much more feasible to train convolutional neural networks (CNNs), a powerful deep learning architecture developed by LeCun et al. [26]. Before the development of cheap GPUs, training CNNs was too computationally expensive for extensive experimentation.

The wide availability and development of deep learners have produced some important changes in the classical pattern recognition framework. The traditional workflow is a three-step process involving preprocessing/transformation, feature extraction, and classification [13], and most research following this paradigm has focused on improving each of these steps. The feature extraction step, for instance, has evolved to such a point that many researchers now view it as a form of feature engineering, the goal being to develop powerful feature vectors calculated to describe patterns in specific ways relevant to the task at hand. These engineered features are commonly described in the literature as handcrafted or handmade features. The main objective behind feature engineering is to create features that place patterns belonging to the same class close to each other in the

*Correspondence: carlos.sillajr@gmail.com
[3]Pontifícia Universidade Católica do Paraná, Rua Imaculada Conceição, 1155, Curitiba, 80215-901, Brazil
Full list of author information is available at the end of the article

feature space, while simultaneously maximizing their distance from other classes.

With the ability to explore more easily and extensively deep learning approaches, autonomous representation learning has gained more attention. With deep learning, the classification scheme is developed in such a way that the classifier itself learns during the training process the best features for describing patterns. In addition, due to the nature of some deep architectures, such as CNN, the patterns are commonly described as an image at the beginning of the process. This has motivated researchers using CNNs in audio classification to develop methods for converting an audio signal into a time-frequency image.

The approach we take in this paper expands previous studies where deep learning approaches are combined with ensembles of texture descriptors for audio classification. Different types of audio images (spectrograms, harmonic and percussion images, and ScatNet scattering representations) are extracted from the audio signal and used for training/fine-tuning CNNs and for calculating the texture descriptors.

Our main contributions to the community are the following:

- For several animal audio classification problems, we test the performance obtained by fine-tuning different pretrained CNNs (AlexNet, GoogleNet, Vgg-16, Vgg-19, ResNet, and Inception) on ImageNet, demonstrating that an ensemble of different fine-tuned CNNs maximizes the performance in our tested animal audio classification problems;
- A simple CNN is trained (not fine-tuned) directly using the animal audio datasets and fused with the ensemble of fine-tuned CNNs.
- Exhaustive tests are performed on the fusion between an ensemble of handcrafted descriptors and an ensemble system based on CNN.
- All MATLAB source code used in our experiments will be freely available to other researchers for future comparisons at https://github.com/LorisNanni.

Extensive experiments on the above approaches and their fusions are carried out on different benchmark databases. These experiments were designed to compare and maximize the performance obtained by varying combinations of descriptors and classifiers. Experimental results show that our proposed system outperforms the use of handcrafted features and individual deep learning approaches.

The remainder of this work is organized as follows: In Section 2, we describe some of the most important works available in the literature regarding deep learning on audio classification tasks, and pattern recognition techniques on animal classification. In Section 3, we describe the method proposed here. In Section 4, we present some details about the CNN architectures used in this work. In Section 5, we portray some facts about the experimental setting. In Section 6, we describe the experimental results, and finally, the conclusions are presented.

2 Related works

To the best of our knowledge, the use of audio images in deep learners started in 2012 when Humphrey and Bello [22] started exploring deep architectures as a way of finding new alternatives that addressed some music classification problems, obtaining state of the art using CNN in automatic chord detection and recognition [23]. In the same year, Nakashika et al. [32] performed music genre classification on the GTZAN dataset [57] starting from spectrograms using CNN applied on feature maps made with the Gray Level Co-occurrence Matrix (GLCM) [19]. One year later, Schlüter and Böck [48] performed music onset detection using CNN, obtaining state of the art at this task. Gwardys and Grzywczak [18] performed music genre classification on the GTZAN dataset using the CNN model winner of the Large Scale Visual Recognition Challenge (ILSVRC) 2012 edition, which was trained on a dataset with more than one million images. Sigtia and Dixon [51] assessed music genre classification on both the GTZAN and ISMIR 2004 datasets. In that paper, the authors offered a suggestion for adjusting CNN parameters to obtain a good performance both in terms of accuracy and time consumption. Finally, Costa et al. [11] performed better than the state of the art on the Latin Music Database (LMD) [52] by using a late fusion strategy to combine CNN classifiers with features from local binary pattern (LBP) and support vector machine (SVM).

While most work using deep learning approaches focus on improving the classification performance, there is also research that focuses on different aspects of the process. Examples of such research include the work of Pons and Serra [45], who point out that most research using CNNs for music classification tasks employ traditional architectures that come from the image processing domain and that employ small rectangular filters applied to spectrograms. Pons and Serra proposed a set of experiments exploring filters of different sizes; however, results proved inferior to the best known classification methods that used handcrafted features for the tested dataset. Wang et al. [59] proposed a novel CNN they called a sparse coding CNN that addressed the problem of sound event recognition and retrieval. In their experiments, they compared their approach against other approaches using 50 of the 105 classes of the Real World Computing Partnership Sound Scene Database (RWCP-SSD). The authors obtained competitive and sometimes superior results compared to most other approaches when evaluating the performance under noisy and clean conditions.

Oramas et al. [43] focused on combining different modalities (album cover images, text reviews, and audio tracks) for multilabel music genre classification using deep learning approaches appropriate for each modality. In their experiments, they verified that the multimodal approach outperformed single modal approaches. Finally, Lim and Lee [27] have proposed a method that uses a convolutional auto-encoder method to perform harmonic and percussive source separation. In another application domain, we also can find some works focused on speech recognition that have been accomplished using CNN as well [21, 30].

Some of the methods used in this paper are based on research that has explored audio classification using a visual time-frequency representation of the sound, which has been explored in different application domains. Research along this line began in 2011, when Costa et al. [8] published results on music genre classification using GLCM to describe texture features extracted from spectrograms that were fed into a SVM. The experiments were conducted on the LMD dataset, and the results were comparable to the state of the art at that time. One year later, Costa et al. [10] assessed music genre classification once again by taking features from spectrogram images, but this time, the authors used more current state-of-the-art texture descriptors, such as LBP [41], which trained SVM classifiers on two music databases, LMD and ISMIR 2004 [6]. Results proved superior to the state of the art on the LMD database. In 2013, Costa et al. [9] used the same strategy with texture features obtained with Local Phase Quantization (LPQ) [42] and Gabor filters [17]. Nanni et al. [37] then experimentally compared several different texture descriptors and ensembles of texture descriptors to find the best general ensemble of classifiers for music genre classification. Montalvo et al. [31] assessed automatic spoken language identification using a similar experimental protocol, starting from spectrograms.

In 2015, some of the same image-based techniques mentioned above were applied to the task of animal classification. Lucio and Costa [28], for instance, performed bird species classification using spectrograms. After that, Freitas et al. [16] used spectrograms to detect North Atlantic right whale calls from audio recordings collected underwater. Nanni et al. [38] performed bird species identification by combining features obtained in the visual domain (spectrograms) with features obtained directly from the audio signal. In the same vein, Nanni et al. [33, 39] performed bird species classification and North Atlantic right whale call identification. In all of these cases, the authors obtained results comparable to the state of the art if not better than that of the state of the art.

The use of non-invasive artificial intelligence techniques based on audio, image, and video data is ideal for identifying and monitoring different types of animal species. These approaches are classified as having an A degree of invasiveness according to the Canadian Council on Animal Care (CCAC[1]) scale of invasiveness (and subsequently pain scale), as they are indirect monitoring techniques. In the related literature, it is possible to find other works where different techniques are used to identify and/or monitor different types of species such as birds [1, 12], whales, frogs [1], and bats [12]. However, most existing works still rely on traditional machine learning approaches, where one needs to use the feature extraction approach, clearly telling the algorithms which engineered features will be used to represent the data.

In this paper, we explore the use of deep learning approaches, specifically approaches based on the convolutional neural network (CNN), a deep learner that is able to automatically learn features directly from the dataset while training. It should be noted that other researchers have also used deep learning-based approaches to deal with different animal classification problems. For example, Branson et al. [4] performed experiments with a CNN for fine-grained classification of bird images. In their experiments with SVM and CNN extracted features, they were able to reduce the error rate on the Caltech-UCSD Birds-200-2011 dataset (CUB-200-2011) [58] (that contains 200 bird species and 11,788 images) by 30% in relation to the technique Part-based One-vs-One Features (POOF) [3].

There are also some works that combine the use of a deep learning approach with other approaches. Cao et al. [7], for instance, combined a CNN with handcrafted features to classify marine animals (fishes and benthic animals). Their experimental results showed that, by combining handcrafted features with CNN learned features, it was possible to achieve better classification results. Salamon et al. [46] investigated the use of combining deep learning (using CNN) and shallow learning for the problem of bird species identification. They employed 5428 bird flight calls from forty-three bird species. In their experiments, they used a Mel-Frequency Cepstral Coefficient (MFCC) approach as baseline, which was surpassed by both approaches. Their best result was obtained by using the combined approach. In [61], the authors used visual, acoustic, and learned features to perform bird species classification, on a dataset composed of bird sounds taken from 14 different species. The authors compared the results individually obtained with these three kinds of feature, with those obtained by combining them using a late fusion strategy. Finally, the best result was obtained by combining visual, acoustic, and learned features, which suggests that there is a complementarity between these different representations.

[1] https://www.ccac.ca/Documents/Standards/Policies/Categories_of_invasiveness.pdf

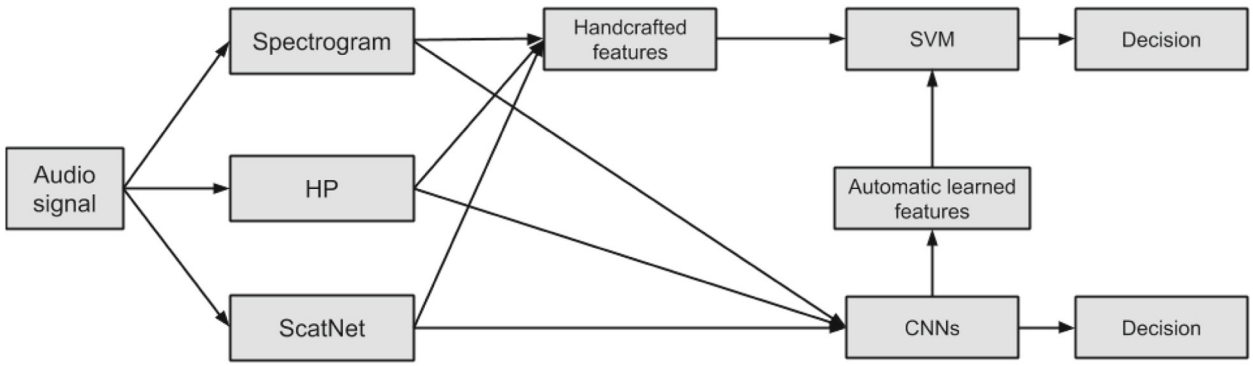


Fig. 1 Overview of the base classifiers

3 Proposed approach

An overview of the base classifiers used in our proposed approach is presented in Fig. 1. The main idea behind our approach is to perform the ensemble of different types of approaches. These approaches can be trained using different types of input. Figure 1 illustrates the different types of input that are used to train the classifiers.

The main idea is that we take an animal audio signal and transform it into a visual image. Different methods can be used to create this image, such as spectrograms (Section 3.2.1), harmonic-percussive spectrogram images (Section 3.2.2), and scattergrams (Section 3.2.3). These images generated from the audio can then be used in one out of two ways. In the first way, different sets of handcrafted features are extracted from the visual representations of the audio and used to train and test a SVM classifier. In the second way, the visual representation of the audio is fed directly to a standard convolutional neural network (CNN), which automatically learns a feature representation. This representation learned by the CNN can be used to train a SVM classifier or to make a decision with the CNN itself. We also extract some acoustic features from the audio signal and train a SVM classifier as a baseline approach.

3.1 Acoustic features

The acoustic features extracted from an audio signal and combined in the tested ensembles are those used in [36] and summarized in Table 1.

In the next section (Section 3.2), we present details about audio image representation.

3.2 Audio image representation

As illustrated in Fig. 2, audio signals are transformed into four different audio images. In this section, we describe the process of transforming audio signals into images.

3.2.1 Spectrogram images

Audio signals are converted into spectrogram images that shows the spectrum of frequencies along the vertical axis as they vary in time along the horizontal axis (shown in Fig. 2a). The intensity of each point in the image represents the signal's amplitude. The audio sample rate is 22,050 Hz, and spectrograms are generated using the Hanning window function with the Discrete Fourier Transform (DFT) computed with a window size of 1024 samples. The left channel is discarded since no considerable difference exists between the content of the left/right audio channels. Spectrogram images undergo a battery of tests to find complementarity among the different representations; a process that led us to select three different values of the lower limit of the amplitude: −70 dBFS, −90 dBFS, and −120 dBFS. At this point, it is important to highlight that as bigger the lower limit value as higher the contrast in the spectrogram image. Thus, we train three different classifiers, one for each of the images using the selected values. The classifiers are combined by sum rule.

3.2.2 Harmonic and percussion images

The harmonic and percussion images are produced using the Harmonic-Percussive Sound Separation (HPSS) method proposed by Fitzgerald [15]. This method works by using a median filter across successive windows of the spectrogram of the audio signal. The harmonic and percussion images are generated using two masks: (1) one generated by performing median filtering across the frequency bins (this enhances the percussive events and suppresses the harmonic components) and (2) one generated by performing median filtering across the time axis (this suppressed the percussive events and enhances the harmonic components). These median filtered spectrograms are applied to the original spectrogram as masks to separate the harmonic and percussive parts of the signal. In this work, we used the Librosa [29] implementation of the HPSS method. The rationale behind the use of these kind

Table 1 Acoustic and visual handcrafted features

Features	Descriptors	Reference
Acoustic	Statistical Spectrum Descriptors (SSD) is a set of statistical measures that describe audio content taken from the moments on the Sonogram (the Sone) of each of the twenty-four critical bands defined according to the Bark scale.	[49]
	Rhythm Histogram (RH) is a feature set where the magnitudes of each modulation frequency bin of the twenty-four critical bands defined according to the Bark scale are summed up to form a histogram of "rhythmic energy" per modulation frequency.	[49]
	Modulation Frequency Variance Descriptor (MVD) is a 420-dimensional feature vector that measures variation over the critical frequency bands for each modulation frequency.	[49]
	Temporal Statistical Spectrum Descriptor (TSSD) is a feature set that incorporates temporal information from the SSD (timbre variations, changes in rhythm, etc.).	[14, 44]
	Temporal Rhythm Histograms (TRH) is a feature set that captures rhythmic changes in music over time.	[49]
Visual	The multiscale uniform local binary pattern (LBP).	[41]
	The multiscale LBP histogram Fourier descriptor (LHF) obtained from the concatenation of LBP-HF.	[63]
	The multiscale rotation invariant co-occurrence of adjacent LBPs (LBP-RI).	[40]
	The Multiscale Local Phase Quantization (MLPQ).	[42]
	Ensemble of LPQ, where different configurations of LPQ are examined.	[35]
	The Heterogeneous Auto-Similarities of Characteristics (HASC) descriptor that is applied to heterogeneous dense features maps.	[47]
	Ensemble of variants of the LHF.	[34]
	The Gabor filter feature extraction method where several different values for scale level and orientation are experimentally evaluated.	[17]
	Extracts the standard Binarized Statistical Image Features (BSIF) by projecting subwindows of the entire image onto subspaces.	[24]
	Adaptive hybrid pattern (AHP), which is an LBP variant that is noise robust because a quantization algorithm is applied that uses an equal probability quantization to maximize partition entropy.	[65]
	Locally Encoded Transform feature histogram (LETRIST) that explicitly encodes the joint information within an image across feature and scale spaces.	[54]
	CodebookLess Model, which is a dense sampling approach similar to Bag of Features (BoF).	[60]

of images is that in some audio classification tasks, the harmonic and the percussive content may have different behavior for different classes considered in the problem. Examples of harmonic and percussion images are shown respectively in Fig. 2b, c.

3.2.3 Scattergram

The scattergram is a representation built from the Scattering Network (ScatNet). This produces an image that is the visualization of the second-order, translation-invariant scattering transform of 1D signals. ScatNet is a wavelet convolutional scattering network [5, 50]. It has achieved state-of-the-art results in many image recognition and music genre recognition challenges. ScatNet resembles a CNN in that the scattering transform is the set of all paths that an input signal might take from layer to layer, but the convolutional filters are predefined as wavelets requiring no learning. Each layer in ScatNet is the association of a linear filter bank wavelet operator (Wop) with a nonlinear operator: the complex modulus. Each operator Wop $1+m$ (m is the maximal order of the scattering transform) performs two operations resulting in two outputs: (1) an energy averaging operation by means of a low-pass filter according to the largest scale, ϕ, and (2) energy scattering operations along all scales using band-pass filters ψ_j with j the scale index.

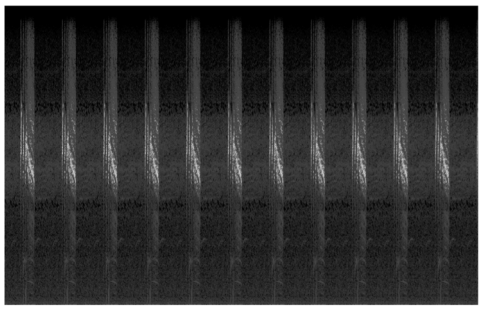
In audio processing the linear operators are constant-Q filter banks. Two layers are typically sufficient for capturing the majority of the energy in an audio signal with an averaging window less than 1 s. The scattering operators rely on a set of built-in "wavelet factories" that are appropriate for specific classes of signals. Wavelets are built by dilating a mother wavelet ψ by a factor $2^{\frac{1}{Q}}$ for some quality factor Q to obtain the filter bank:

$$\psi_j(t) = 2^{\frac{-j}{Q}}\psi(2^{\frac{-j}{Q}}t). \tag{1}$$

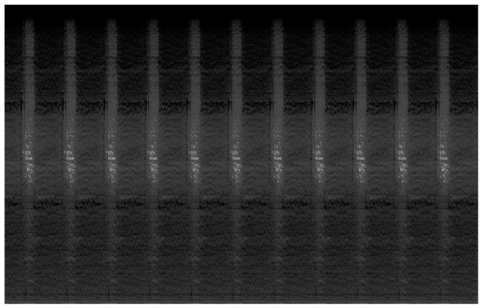
The mother wavelet ψ is chosen such that adjacent wavelets barely overlap in frequency. The scattering coefficients are defined by:

$$S_1x(t,j_1) = |x \star \psi_{j1}| \star \phi(t) \tag{2}$$

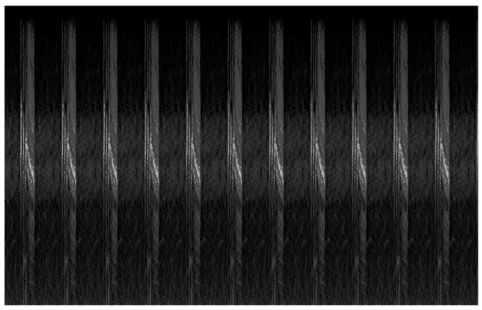
$$S_2x(t,j_1,j_2) = x \star \psi_{j1}| \star \psi_{j1}| \star \phi(t), \tag{3}$$



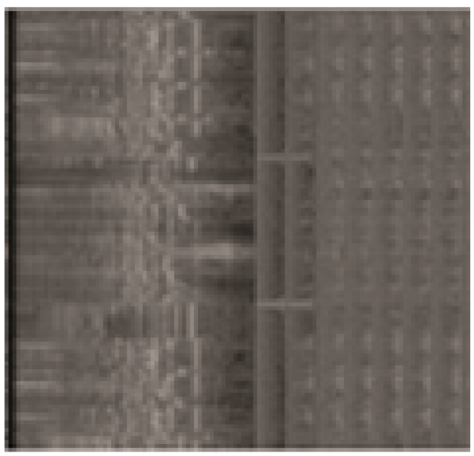
(a) Spectrogram.



(b) Harmonic.



(c) Percussion.



(d) Scattergram.

Fig. 2 Four types of audio images extracted from the audio signals. In all representation, the horizontal axis regards to time, and the vertical axis regards to frequencies

and so on.

The scattering representation S is a cell array, whose elements correspond to respective layers in the scattering transform.

In this work, we use the MATLAB toolbox ScatNet to generate the audio scattergrams. This toolbox is available at http://www.di.ens.fr/data/software/scatnet/. More details about the inner workings of the scattergram are available at [2].

3.2.4 Visual feature extraction

Visual feature extraction is a three-step process:

- Step 1: An audio signal is transformed into four types of audio images (see Section 3.2 for details): (i) spectrogram, (ii) percussion, (iii) harmonic images, and (iv) scattergram.
- Step 2: Each image is divided into subwindows, i.e., it is divided into three zones along the x-axis. By this way, the visual descriptors are applied on these non-overlaping zones, which regard to different moments of the audio signal.
- Step 3: Sets of handcrafted texture descriptors are extracted from the subwindows, with each type of descriptor classified using a separate SVM. In addition, different CNNs are tuned/trained using the audio images (see Section 4 for details).

The handcrafted features combined with each other and ensembles of CNNs are those tested in [36] and listed in Table 1. As the focus of this paper is on CNN, the reader is referred to [36] or to the original references for more details.

4 Convolutinal neural networks

In this section, we describe each step using CNN for feature extraction and/or classification. CNNs are deep feed-forward neural networks (NNs) composed of interconnected neurons that have inputs with learnable weights, biases, and activation functions. CNNs are built by repeatedly concatenating five classes of layers: convolutional (CONV), activation (ACT), and pooling (POOL), which are followed by a last stage that typically contains fully connected (FC) layers and a classification (CLASS) layer. The CONV layer performs feature extraction by convolving input to filters. After each CONV layer, a non-linear ACT layer is applied, such as the non-saturating ReLU (rectified linear unit) function $f(x) = max(0,x)$ or the saturating hyperbolic tangent $f(x) = tanh(x)$, $f(x) = |tanh(x)|)|$, or the sigmoid function $f(x) = (1+e^{-x})^{-1}$. Non-linearity activation is useful to improve classification and the learning capabilities of the network. POOL layers perform non-linear downsampling operations aimed at reducing the spatial size of the representation while simultaneously decreasing (1) the number of parameters,

(2) the possibility of overfitting, and (3) the computational complexity of the network. It is a common practice to insert a POOL layer between CONV layers. Typical pooling functions are max and average. FC layers have neurons that are fully connected to all the activations in the previous layer and are applied after CONV and POOL layers. In the higher layers, multiple FC layers and one CLASS layer perform the final classification. A widely used activation function in the CLASS layer is SoftMax.

For audio classification, the audio images are downsized in order to speed up CNN classification performance [11]. Downsizing images reduces the number of neurons in the convolutional layers as well as the number of trainable parameters of the network. Downsizing is accomplished by taking only the first pixel of every four pixels in 2×2 subwindows of the image. As a result, both image height and width are cut by half.

The CNN used in this work (see Fig. 3) has two 2D convolutional layers with 64 filters followed by a max-pool layer. The 5th layer is a fully connected layer with 500 neurons. The activation function is the rectified linear units (ReLUs), except for the neurons of the last layer, which use Softmax, as mentioned above. It is important that the number of neurons in the last layer equals the number of classes for each problem. Training is performed using backpropagation with 50 epochs. Once trained, the output of the 5th layer is used for feature extraction. This produces a 500-dimensional vector image representation.

Fine-tuning a CNN essentially restarts the training process of a pretrained network so that it learns a different classification problem. We fine-tune CNNs that have already been pretrained (initialized) on natural image data (illustrated in Fig. 4). Each of the fine-tuned CNNs is then used in two ways: (1) as an image feature extractor, which results in a feature vector extracted from the image (after that, these vectors are used to train and test multiclass support vector machines (SVMs)), and (2) as a classifier, generating SoftMax probabilities. The posterior probabilities from the ensemble of SVMs and SoftMax classifiers are used to determine the class of an image.

We fine-tune the weights of the pretrained CNN by keeping the earlier CONV layers of the network fixed and by fine-tuning only the higher-level FC layers since these layers are specific to the details of the classes contained in the target dataset. The last layer is designed to be the same size as the number of classes in the new data. All the FC layers are initialized with random values and trained from scratch using the backpropagation algorithm with data from new target training set. The tuning procedures is performed using 40 epochs, a mini-batch with 10 observations at each iteration, and learning rate of $1e-4$.

In this work, we test and combine different CNN architectures:

1. *AlexNet* [25]. This CNN is the winner of the ImageNet ILSVRC challenge in 2012 and has proven to be quite popular. AlexNet is composed of both stacked and connected layers. It includes five CONV layers followed by three FC layers, with some max-POOL layers inserted in the middle. A rectified linear unit non-linearity is applied to each convolutional along with a fully connected layer to enable faster training.
2. *GoogleNet* [56]. This CNN is the winner of the ImageNet ILSVRC challenge in 2014. It introduces a new "Inception" module (INC), which is a subnetwork consisting of parallel convolutional filters whose outputs are concatenated. INC greatly reduces the number of parameters, much lower than AlexNet, for example. GoogleNet is composed of 22 layers that require training and five POOL layers.
3. *VGGNet* [53]. This CNN placed second in ILSVRC 2014. It is a very deep network that includes 16

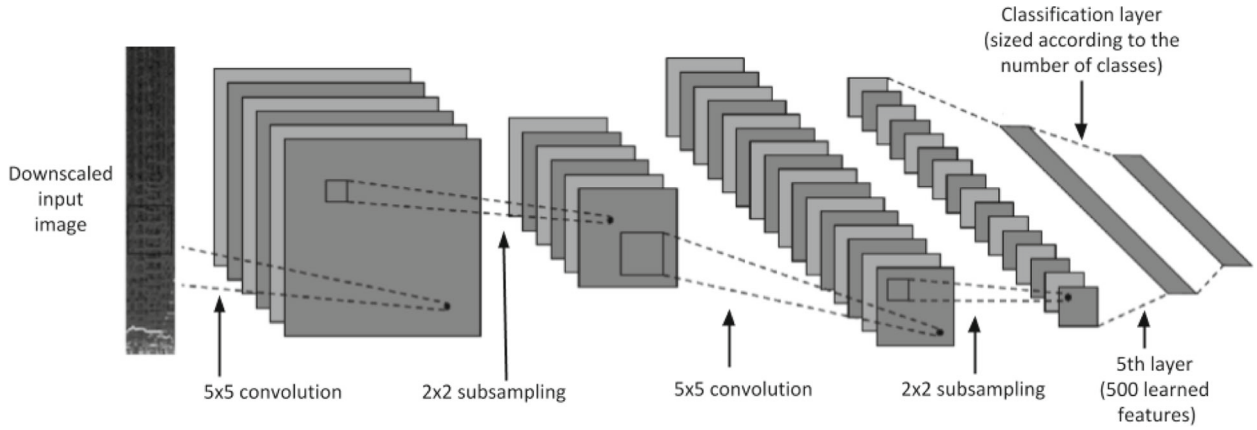


Fig. 3 The deep convolutional neural network architecture (adapted from [11])

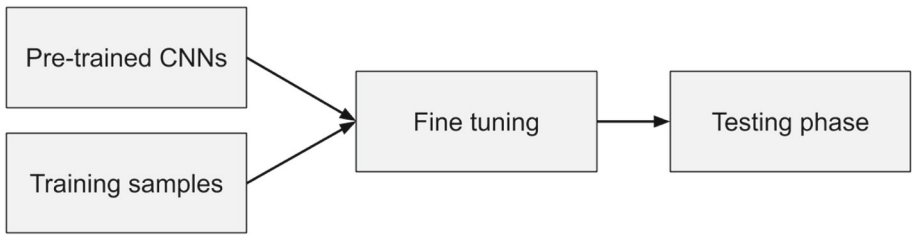


Fig. 4 Example of fine tuning

CONV/FC layers. The CONV layers are extremely homogeneous and use very small (3×3) convolution filters with a POOL layer inserted after every two or three CONV layers (instead after each CONV layer as in AlexNet). The two best-performing VGG models (Vgg-16 and Vgg-19), with 16 and 19 weight layers, respectively, are available as pretrained models.

4. *ResNet* [20]. This CNN is the winner of ILSVRC 2015. ResNet is a network that is approximately twenty times deeper than AlexNet and eight times deeper than VGGNet. The main novelty of this CNN is the introduction of residual (RES) layers, making it a kind of "network-in-network" architecture, which can be treated as a set of "building blocks" to construct the network. It uses special skip connections and batch normalization. The FC layers at the end of the network are substituted by global average pooling. ResNet explicitly reformulates layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions. ResNet is much deeper than VGGNet, but the model size is smaller and easier to optimize than is the case with VGGNets.
5. *InceptionV3*. This is a recent CNN topology that was proposed in [55]. The networks in InceptionV3 are scaled up networks to utilize computation as efficiently as possible. This is accomplished by suitable factorized convolutions and aggressive regularization. As a result, the computational cost of Inception is lower than even ResNet.

5 Experimental settings

In this section, we describe details about the datasets used in this work and about the classifiers and ensembles used here.

5.1 Datasets

Our proposed approach is assessed using the recognition rate (i.e., accuracy or AUC-ROC, depending on the dataset) as the performance indicator on the following animal audio datasets using:

5.1.1 BIRD

The Bird Songs 46 dataset [28] that is freely available and developed as a subset used in [38]. All bird species with less than ten samples were removed to build this subset. This dataset is composed of 2814 audio samples of bird vocalization taken from 46 different species found in the South of Brazil. Although the Bird Songs 46 dataset is composed exclusively of bird songs, calls related to other bird species are sometimes heard in the background. The protocol used for this dataset is a stratified 10-fold cross-validation strategy.

5.1.2 BIRDZ

The control and real-world audio dataset used in [64]. This dataset is composed of field recordings of eleven bird species taken from the Xeno-canto Archive and was selected because it lends itself to comparison. BIRDZ contains 2762 bird acoustic events (11 classes) with 339 detected "unknown" events corresponding to noise and other unknown species vocalizations.

5.1.3 WHALE

The whale identification dataset used in "The Marinexplore and Cornell University Whale Detection Challenge." WHALE is composed of 84,503 audio clips that are 2 s long and that contain mixtures of right whale calls, nonbiological noise, and other whale calls. Thirty thousand samples have class labels. We used 20,000 samples for the training set and the remaining 10,000 samples for the testing set. The results on this dataset are described using the area under the receiver operating characteristic (ROC) curve (AUC), because it is the performance indicator used in the original whale detection challenge.

5.1.4 BAT

A dataset for tree classification from bat-like echolocation signals shared by Yovel et al. [62]. BAT contains 1000 patterns for each of the following four classes: Apple tree (*Malus sylvestris*), Norway spruce tree (*Picea abies*), Blackthorn tree (*Prunus spinosa*), and Common beech tree (*Fagus sylvatica*). The dataset is built by a biomimetic sonar system that has a sonar head with three trans-

Table 2 Details of fine-tuned CNNs

Network	Depth	Size	Parameters (in millions)	Image input size
AlexNet	8	227 MB	61.0	227-by-227
Vgg-16	16	515 MB	138	224-by-224
Vgg-19	19	535 MB	144	224-by-224
GoogleNet	22	27 MB	7.0	224-by-224
InceptionV3	48	89 MB	23.9	299-by-299
ResNet50	50	96 MB	25.6	224-by-224

ducers that create and record the vegetation echoes. For each tree, the echoes are recorded from different angles thus allowing us to classify the trees independently from the aspect angle. As in [62], the recorded echoes are preprocessed as follows:

1. The echo regions are cut out from the recorded signal in the time domain and are transformed into the time-frequency space by calculating the magnitude of their spectrograms.
2. The Hann window (with 80% overlap between sequential windows) is used to calculate the spectrograms.
3. A denoising technique is performed to reduce the noise and enhance the quality of the signal. Each echo is represented by spectrogram composed by 85 (frequency bins) $\times 160$ (time bins).

The protocol used for this dataset is a stratified fivefold cross-validation strategy.

5.2 SVM configuration

Sets of these features are classified using separate SVMs, with results combined for a final ensemble decision. The SVM parameters were not optimized aiming to avoid the risk of overfitting. In this way, the C parameter was set to 1000 and γ was set to 0.1 in all experiments. Before the classification step, the features are linearly normalized to [0, 1], and the Radial Basis Function (RBF) kernel was used to perform the SVM training. In addition, CNNs (the focus of this paper) are tuned/trained using the audio images. Ensembles of CNNs and handcrafted features are then tested to maximize generalizability and performance.

The SVM used in our experiments is the one-versus-all SVM. Features are linearly normalized to [0, 1] before classification, and SVMs are combined by sum rule, with the final ensemble decision for a given sample x being the class that receives the largest support, defined as:

$$\text{sum}(x) = \arg\max_{k=1}^{c} \sum_{i=1}^{n} P(\omega_k | y_i(x)) \tag{4}$$

in which x is the instance to be classified, c is the number of classes, n is the number of classifiers in the ensemble, y_i is the label predicted by the ith classifier in a problem with the following class labels $\Omega = \omega_1, \omega_2, ..., \omega_c$, and $P(\omega_k | y_i(x))$ is the probability of the sample x belonging to class ω_k according to the ith classifier.

5.3 Deep learning configuration

One application of deep learning we tested is a model trained from scratch. This model is illustrated in Fig. 3. The fine-tuned models we used are listed in Section 4, and their details are presented in Table 2.

5.4 Ensemble configuration

In the experiments, we have employed ensembles of different fine-tuned CNNs using different audio images. Figure 5 presents an overview of this approach. The idea is

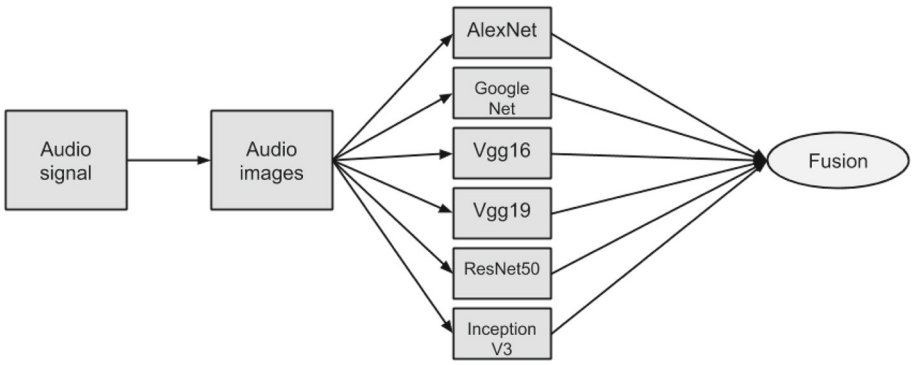


Fig. 5 Fusion of the sets of CNNs

that each fine-tuned deep neural network is trained using the same visual image as input. The final classification is given by the sum rule.

The naming convention used hereafter for each ensemble is the following:

- Fus_Spec: Ensemble of the six fine-tuned CNNs using the spectograms as audio images.
- Fus_HP: Ensemble of the six fine-tuned CNNs using the harmonic percussive as audio images.
- Fus_Scatter: Ensemble of the six fine-tuned CNNs using the scattergram as audio images.
- Fus_Hand: Ensemble of the handcrafted features presented in Table 1.

6 Results and discussion

Table 3 presents the results obtained using different approaches. In this section, we will perform different analyses of the results in order to answer the following research questions:

RQ1 What is the performance of the fine-tuned deep learning approaches in comparison with the handcrafted features?

RQ2 What is the performance of the fine-tuned deep learning approaches in comparison with the standard CNN?

RQ3 Does the different fine-tuned deep learning approaches perform similarly across the different

Table 3 Performance of different approaches on each animal sound dataset

Approach	Descriptor	BIRD	BIRDZ	WHALE[†]	BAT
Handcrafted features with SVM	Acoustic features	80.2	82.1	85.8	–
	LBP	85.8	87.0	90.6	91.2
	LBP-HF	85.0	86.2	89.9	92.6
	LBP-RI	86.1	87.5	91.0	93.0
	MLPQ	87.5	88.8	92.1	93.5
	HASC	87.9	89.1	92.0	92.9
	LHF	86.0	86.9	90.5	91.9
	GABOR	87.3	87.2	90.3	90.9
	BSIF	88.8	87.5	90.4	92.4
	AHP	84.4	77.5	89.9	92.1
	LETRIST	67.7	75.6	90.3	89.5
	BoF	89.9	60.4	87.2	94.2
Deep learning using the four types of audio images	CNN (Fig. 3)	61.8	84.4	93.5	98.6
	AlexNet	79.8	88.9	95.5	97.8
	GoogleNet	77.8	86.1	94.8	95.9
	Vgg-16	83.6	90.4	96.6	90.1
	Vgg-19	86.3	89.6	96.6	88.6
	ResNet50	81.9	88.9	96.1	93.7
	InceptionV3	82.3	88.5	96.5	85.9
Ensembles of deep learning	Fus_Spec	87.9	91.0	96.6	97.3
	Fus_HP	49.8	88.1	95.2	–
	Fus_Scatter	46.6	91.3	96.7	–
	Fus_Spec + Fus_HP + Fus_Scatter	87.2	93.9	97.1	97.3*
	Fus_Spec + Fus_Scatter	87.9	94.8	**97.2**	97.3*
	Fus_Spec + Fus_Scatter + CNN	84.0	95.1	96.1	98.7*
Ensembles of DL and handcrafted	Fus_Spec + Fus_Scatter + CNN + Fus_Hand	94.1	**99.0**	95.9	**99.3***
	Fus_Spec + Fus_Scatter + Fus_Hand	94.7	98.9	96.5	98.9*
Related works	Deep learning, acoustic, and visual features [36]	**94.8**	–	93.3	–
	Acoustic and visual features [39]	94.5	–	92.2	–
	MFCC + SVM [64]	–	93.6	–	–
	DFT + SVM [62]	–	–	–	92.0

The rates are described using accuracy, except for the WHALE dataset, in which the rates are in AUC-ROC

[*] Fus_Scatter and Fus_HP were not used in this result once they were not available for BAT

[†] The metric used for the WHALE dataset is AUC-ROC

audio animal datasets? Or are there approaches that perform particularly well for each dataset?

RQ4 Is it possible to improve the obtained results by using different types of ensembles of the different approaches?

RQ5 How does the proposed ensembles compare with results reported in the literature?

In order to have a general feeling about the different approaches, we have used the ranking principle from the Friedman statistical test to compare the different approaches under the different datasets. Table 4 presents the approaches ordered by their average rankings across the four datasets. The approaches which were unable to be applied to the BAT dataset were not considered in the rankings.

In relation to RQ1, if we analyze the results from the ranking of the different approaches across the different animal audio datasets, the handcrafted approaches HASC (11) and MLPQ (11.5) obtain better average rankings, 11 and 11.5, respectively, than Vgg-19 (11.5), AlexNet (11.875), Vgg-16 (12.5), ResNet50 (12.5), Inception V3 (15.625), and GoogleNet (15.75). BSIF, LBP-RI, and BoF obtained better rankings than Inception V3 (15.625) and GoogleNet (15.75). It should be noted that these results were obtained by considering only the approaches that have performance scores in all four datasets. Overall, all deep learning approaches obtained worse results than most handcrafted approaches in the BIRD dataset, which make their average ranks drop in comparison with the handcrafted features.

Table 4 Ordered rankings of the approaches

Algorithm	Average ranking
Fus_Spec + Fus_Scatter + Fus_Hand	2.875
Fus_Spec + Fus_Scatter + CNN + Fus_Hand	3.500
Fus_Spec + Fus_Scatter	4.500
Fus_Spec	5.750
Fus_Spec + Fus_HP + Fus_Scatter	6.000
Fus_Spec + Fus_Scatter + CNN	7.875
HASC	11.000
MLPQ	11.500
Vgg-19	11.500
AlexNet	11.875
Vgg-16	12.500
ResNet50	12.500
BSIF	13.375
LBP-RI	13.875
BoF	15.250
CNN	15.500
InceptionV3	15.625
GoogleNet	15.750
GABOR	16.375
LBP	16.750
LHF	16.750
LBP-HF	17.875
AHP	19.375
LETRIST	22.125

Regarding RQ2, Vgg-19, AlexNet, Vgg-16, and ResNet50 obtained better average rankings than the standard CNN approach (average ranking 15.5), while InceptionV3 and GoogleNet obtained slightly lower rankings 15.625 and 15.75, respectively. Our analysis of this result is that the fine-tuned CNNs obtained at least similar performance in compasion with the standard CNN.

Considering the performance of the different fine-tuned deep learning approaches across the different datasets (RQ3), the analysis of the average rankings shows that the Vgg-19 performs better than the other fine-tuned deep learning approaches on average. However, the best obtained results for each dataset (considering only the SVM with handcrafted features and the deep learning approaches) are as follows: 89.9% for the BIRD dataset with BoF, 90.4% for the BIRDZ dataset with Vgg-16, 96.6% for the WHALE dataset with Vgg-16 and Vgg-19, and 97.8% for the BAT dataset with AlexNet. As mentioned earlier, most of the deep learning approaches were outperformed by the handcrafted feature sets in the BIRD dataset, but obtained competitive results for the other three datasets.

In order to attempt to improve the results and answer RQ4, we performed the ensemble of different approaches, using the naming convention presented in Section 5.4. The analysis of the average ranking results shows that the best average rank (2.875) was obtained by the ensemble composed of *Fus_Spec* + *Fus_Scatter* + *Fus_Hand*. This is an interesting result that corroborates with our previous results that demonstrated that there is a complementarity between handcrafted and learned features with a CNN in a sound classification task [11]. Another interesting result is that all ensembles outperform (by analyzing the average rankings) the other handcrafted and deep learning approaches in isolation.

In relation to related work (RQ5), with the exception of Vgg-16, the other deep learning approaches outperform the state of the art for the BAT dataset, being the best individual result obtained by the *Fus_Spec* + *Fus_Scatter* + *CNN* + *Fus_Hand*. For the BIRDZ and WHALE datasets, the ensembles of deep learning and handcrafted features outperform the state-of-the-art results. For the BIRD dataset, although the ensembles of deep learning and handcrafted features do not outperform the state of the art, they obtain the best results in the dataset (94.1%

and 94.7%, respectively), which represents an increase of 6.2 and 6.8 percent points in comparison with the best result (87.9%) obtained by the *Fus_Spec* and *Fus_Spec* + *Fus_Scatter* ensembles and an increase of 4.2 and 4.8 percent points in comparison with the best individual results 89.9% obtained with the BoF. The result on this dataset emphasizes, once again, the complementarity between handcrafted and learned features.

Regarding the WHALE dataset, it is important to remark that it was built for a Kaggle competition. Only the training set is available, so we cannot report a fair comparison with the competitors in the contest. The winner of the contest obtained an AUC of 0.984, but it used a larger training set. The winner of the contest combines contrast-enhanced spectrograms, template matching, and gradient boosting. Our aim is to show that an ensemble of descriptors based on CNN transfer learning works very well when used to represent an audio pattern. In the future, we plan on testing our approach for comparing two subwindows of the spectrograms instead of the standard template matching method used by the winner of the Kaggle competition.

All the datasets tested in this paper are freely available and tested here with a clear testing protocol. In this way, we report a baseline performance for the audio classification that can be used to compare other methods developed by future researchers.

7 Conclusion

In this paper, we explored the use of deep learning approaches for automated audio classification. The approaches examined here are based on the convolutional neural network (CNN), a deep learning technique that is able to automatically learn features directly from the dataset during the training process. Different types of audio images (spectrograms, harmonic and percussion images, and ScatNet scattering representations) were extracted from the audio signal and used for calculating the texture descriptors and for training/fine-tuning CNNs. In addition, a simple CNN was trained (not fine-tuned) directly using several different types of audio datasets and fused with the ensemble of fine-tuned CNNs using different pretrained CNNs (AlexNet, GoogleNet, Vgg-16, Vgg-19, ResNet, and Inception) on ImageNet. The experimental results presented in this paper demonstrate that an ensemble of different fine-tuned CNNs maximizes the performance in our tested animal audio classification problems. In addition, the fusion between an ensemble of handcrafted descriptors and an ensemble system based on CNN improved results. Our proposed system was shown to outperform previous state-of-the-art approaches. To the best of our knowledge, this is the largest study on CNNs in audio classification (several topologies in four different datasets).

In the future, we aim to add other datasets to those used in the experiments reported here, in order to obtain a more complete validation of the proposed ensemble. We intend to test this system with different sound classification tasks, as well as different CNN topologies, different parameter settings in the fine-tuning step of transfer learning, and different approaches for data augmentation. We also plan to evaluate strategies to select the region of interest of the spectrograms, aiming to select only the most important subwindow of the full spectrograms.

Finally, we want to highlight the fact that the approach based on the extraction of visual features is freely available to other researchers for future comparisons. MATLAB code is located at https://github.com/LorisNanni.

Abbreviations

ACT: Activation layer; AHD: Adaptive hybrid pattern; AUC: Area under the curve; BSIF: Binarized Statistical Image Features; BoF: Bag of Features; CCAC: Canadian Council on Animal Care; CLASS: Classification layers of a neural network; CONV: Convolutional layer; CNN: Convolutional neural network; CUB-200-2011: Caltech-UCSD Birds-200-2011 dataset; dBFS: Decibels relative to full scale; DFT: Discrete Fourier Transform; FC: Fully connected layer; FUS_Hand: Ensemble of handcrafted features presented in Table 1; FUS_HP: Ensemble of six fine-tuned CNNs using harmonic and percussive spectrogram images; FUS_Scatter: Ensemble of six fine-tuned CNNs using scattergram images; FUS_Spec: Ensemble of six fine-tuned CNNs using spectrogram images; GLCM: Gray Level Co-occurrence Matrix; GPU: GraphicS processing unit; HASC: Heterogeneous Auto-Similarities of Characteristics; HPSS: Harmonic-Percussive Sound Separation; ILSVRC: ImageNet large visual recognition challenge; INC: Inception module from GoogleNet; ISMIR 2004: Database used in the 5th international conference on music information retrieval; GTZAN: GTZAN genre collection database; LBP: Local binary patterns; LBP-RI: Rotation invariant co-occurrence of LBPs; LETRIST: Locally Encoded Transform feature histogram; LHF: Local binary patterns concatenated with fourier descriptor; LMD: Latin Music Database; LPQ: Local Phase Quantization; MFCC: Mel-Frequency Cepstral Coefficients; MLPQ: Multscale Local Phase Quantization; MVD: Modulation Frequency Variance Descriptor; NN: Neural network; POOF: Part-based one-vs-one features; POOL: Pooling layer; RBF: Radial basis function; ReLU: Rectified linear unit; RES: Residual layers from ResNet; ROC: Receiver operating characteristic; RH: Rhythm histogram; RQ: Research question; RWCP-SSD: Real World Computing Partnership Sound Scene Database; ScatNet: Scattering network; SSD: Statistical Spectrum Descriptors; SVM: Support vector machine; TSSD: Temporal Statistical Spectrum Descriptor; TRH: Temporal Rhythm Histograms; Wop: Wavelet operator

Acknowledgements

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research. We also acknowledge the Brazilian Research-support agencies: The National Council for Scientific and Technological Development (CNPq), Coordination for the Improvement of Higher Level Personnel (CAPES), and Araucária Foundation. We thank the anonymous reviewers for providing valuable feedback to improve our manuscript.

Authors' contributions

LN, YMGC, and CNSJ were responsible for the conceptualization of the work. LN, RLA, and RBM designed and performed the experiments. LN, YMGC, and CNSJ were responsible for analyzing the results. LN, YMGC, SB, and CNSJ wrote the first manuscript. Further changes and corrections were executed by YMGC, RLA, and CSNJ. The guidance of the whole work was performed by LN, YMGC, SB, and CNSJ. Moreover, all authors were involved in the investigation, and reviewed and approved the final manuscript.

Funding

This research was partially supported by NVIDIA Corporation and by Brazilian Research-support agencies National Council for Scientific and Technological Development (CNPq), Coordination for the Improvement of Higher Level Personnel (CAPES), and Araucária Foundation.

Availability of data and materials
The datasets supporting the conclusions of this article are available in the internet. The datasets BIRD and BIRDZ were extracted from Xeno-Canto website, https://www.xeno-canto.org/. The WHALE dataset was obtained from the Kaggle website, https://www.kaggle.com/c/whale-detection-challenge/data, and the BAT spectrograms are available at https://github.com/LorisNanni/BATS-spectrograms. The MATLAB codes are available at https://github.com/LorisNanni.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Department of Information Engineering, University of Padua, Viale Gradenigo 6, Padua, Italy. [2]Department of Informatics, State University of Maringá, Av. Colombo 5790, Maringá, 87020-900, Brazil. [3]Pontifícia Universidade Católica do Paraná, Rua Imaculada Conceição, 1155, Curitiba, 80215-901, Brazil. [4]Missouri State University, 901 S. National, Springfield, 65804, USA.

Received: 27 March 2019 Accepted: 24 March 2020
Published online: 26 May 2020

References

1. M. A. Acevedo, C. J. Corrada-Bravo, H. Corrada-Bravo, L. J. Villanueva-Rivera, T. M. Aide, Automated classification of bird and amphibian calls using machine learning: a comparison of methods. Ecol. Inform. **4**(4), 206–214 (2009)
2. J. Andén, S. Mallat, Deep scattering spectrum. IEEE Trans. Signal Process. **62**(16), 4114–4128 (2014). https://doi.org/10.1109/TSP.2014.2326991
3. T. Berg, P. N. Belhumeur, in *2013 IEEE Conference on Computer Vision and Pattern Recognition*. Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation, (2013), pp. 955–962. https://doi.org/10.1109/CVPR.2013.128
4. S. Branson, G. Van Horn, S. Belongie, P. Perona, Bird species categorization using pose normalized deep convolutional nets. arXiv preprint (2014). arXiv:1406.2952
5. J. Bruna, S. Mallat, Invariant scattering convolution networks. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1872–1886 (2013)
6. P. Cano, E. Gómez, F. Gouyon, P. Herrera, M. Koppenberger, B. Ong, X. Serra, S. Streich, N. Wack, *Ismir 2004 audio description contest*. (Music Technology Group of the Universitat Pompeu Fabra, Tech. Rep, 2006)
7. Z. Cao, J. C. Principe, B. Ouyang, F. Dalgleish, A. Vuorenkoski, *Marine animal classification using combined cnn and hand-designed image features*. (IEEE, 2015), pp. 1–6. https://doi.org/10.23919/oceans.2015.7404375
8. Y. M. G. Costa, L. S. Oliveira, A. L. Koerich, F. Gouyon, in *Systems, Signals and Image Processing (IWSSIP) 2011 18th International Conference on*. Music genre recognition using spectrograms (IEEE, 2011), pp. 1–4
9. Y. M. G. Costa, L. S. Oliveira, A. L. Koerich, F. Gouyon, in *Systems, Signals and Image Processing (IWSSIP) 2013 20th International Conference on*. Music genre recognition based on visual features with dynamic ensemble of classifiers selection (IEEE, 2013), pp. 55–58. https://doi.org/10.1109/iwssip.2013.6623448
10. Y. M. G. Costa, L. S. Oliveira, A. L. Koerich, F. Gouyon, J. Martins, Music genre classification using LBP textural features. Signal Process. **92**(11), 2723–2737 (2012)
11. Y. M. G. Costa, L. S. Oliveira, C. N. Silla Jr, An evaluation of convolutional neural networks for music classification using spectrograms. Appl. Soft Comput. **52**, 28–38 (2017)
12. V. I. Cullinan, S. Matzner, C. A. Duberstein, Classification of birds and bats using flight tracks. Ecol. Inform. **27**, 55–63 (2015)
13. R. O. Duda, P. E. Hart, D. G. Stork, *Pattern Classification and Scene Analysis 2nd ed*. (Wiley Interscience, 1995)
14. S. Fagerlund, Bird species recognition using support vector machines. EURASIP J. Adv. Signal Process. **2007**. https://doi.org/10.1155/2007/38637
15. D. Fitzgerald, in *13th International Conference on Digital Audio Effects (DAFx-10)*. Harmonic/percussive separation using median filtering, (2010)
16. G. K. Freitas, R. L. Aguiar, Y. M. G. Costa, in *Computer Science Society (SCCC) 2016 35th International Conference of the Chilean*. Using spectrogram to detect north atlantic right whale calls from audio recordings (IEEE, 2016), pp. 1–6. https://doi.org/10.1109/sccc.2016.7836034
17. D. Gabor, Theory of communication. part 1 The analysis of information. J. Inst. Electr. Eng. Part III: Radio Commun. Eng. **93**(26), 429–441 (1946)
18. G. Gwardys, D. Grzywczak, Deep image features in music information retrieval. Int. J. Electron. Telecommun. **60**(4), 321–326 (2014)
19. R. M. Haralick, Statistical and structural approaches to texture. Proc. IEEE. **67**(5), 786–804 (1979)
20. K. He, X. Zhang, S. Ren, J. Sun, in *Proceedings of the IEEE conference on computer vision and pattern recognition*. Deep residual learning for image recognition, (2016), pp. 770–778. https://doi.org/10.1109/cvpr.2016.90
21. M. Dong, Q. Mao, Y. Zhan, in *Proceedings of the 22Nd ACM International Conference on Multimedia, MM '14*. Speech emotion recognition using CNN (ACM, New York, 2014), pp. 801–804. https://doi.org/http://doi.acm.org/10.1145/2647868.2654984
22. E. J. Humphrey, J. P. Bello, in *Machine Learning and Applications (ICMLA) 2012 11th International Conference on, vol. 2*. Rethinking automatic chord recognition with convolutional neural networks (IEEE, 2012), pp. 357–362. https://doi.org/10.1109/icmla.2012.220
23. E. J. Humphrey, J. P. Bello, Y. LeCun, in *ISMIR*. Moving beyond feature design: Deep architectures and automatic feature learning in music informatics, (2012), pp. 403–408
24. J. Kannala, E. Rahtu, in *Pattern Recognition (ICPR) 2012 21st International Conference on*. Bsif: Binarized statistical image features (IEEE, 2012), pp. 1363–1366
25. A. Krizhevsky, I. Sutskever, G. E. Hinton, in *Advances in Neural Information Processing Systems*. Imagenet classification with deep convolutional neural networks, (2012), pp. 1097–1105. https://doi.org/10.1145/3065386
26. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel, Backpropagation applied to handwritten zip code recognition. Neural Comput. **1**(4), 541–551 (1989)
27. W. Lim, T. Lee, in *Signal Processing Conference (EUSIPCO) 2017 25th European*. Harmonic and percussive source separation using a convolutional auto encoder (IEEE, 2017), pp. 1804–1808. https://doi.org/10.23919/eusipco.2017.8081520
28. D. R. Lucio, Y. M. G. Costa, in *Computing Conference (CLEI) 2015 Latin American*. Bird species classification using spectrograms (IEEE, 2015), pp. 1–11. https://doi.org/10.1109/clei.2015.7359990
29. B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, O. Nieto, in *Proceedings of the 14th Python in Science Conference*. librosa: Audio and music signal analysis in python, (2015), pp. 18–25. https://doi.org/10.25080/majora-7b98e3ed-003
30. V. Mitra, W. Wang, H. Franco, Y. Lei, C. Bartels, M. Graciarena, in *Fifteenth Annual Conference of the International Speech Communication Association*. Evaluating robust features on deep neural networks for speech recognition in noisy and channel mismatched conditions, (2014)
31. A. Montalvo, Y. M. G. Costa, J. R. Calvo, in *Iberoamerican Congress on Pattern Recognition*. Language identification using spectrogram texture (Springer, 2015), pp. 543–550. https://doi.org/10.1007/978-3-319-25751-8_65
32. T. Nakashika, C. Garcia, T. Takiguchi, in *Thirteenth Annual Conference of the International Speech Communication Association*. Local-feature-map integration using convolutional neural networks for music genre classification, (2012)
33. L. Nanni, R. L. Aguiar, Y. M. G. Costa, S. Brahnam, C. N. Silla Jr, R. L. Brattin, Z. Zhao, Bird and whale species identification using sound images. IET Comput. Vis. (2017). https://doi.org/10.1049/iet-cvi.2017.0075
34. L. Nanni, S. Brahnam, A. Lumini, Combining different local binary pattern variants to boost performance. Expert Syst. Appl. **38**(5), 6209–6216 (2011)
35. L. Nanni, S. Brahnam, A. Lumini, T. Barrier. Ensemble of Local Phase Quantization Variants with Ternary Encoding (Springer, Berlin Heidelberg, 2014), pp. 177–188. https://doi.org/10.1007/978-3-642-39289-4_8
36. L. Nanni, Y. M. G. Costa, R. L. Aguiar, C. N. Silla Jr, S. Brahnam, Ensemble of deep learning, visual and acoustic features for music genre classification. J. New Music Res., 1–15 (2018). https://doi.org/10.1080/09298215.2018.1438476

37. L. Nanni, Y. M. G. Costa, S. Brahnam, in *22nd International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*. Set of texture descriptors for music genre classification, (2014)
38. L. Nanni, Y. M. G. Costa, D. R. Lucio, C. N. Silla Jr., S. Brahnam, in *Tools with Artificial Intelligence (ICTAI) 2016 IEEE 28th International Conference on*. Combining visual and acoustic features for bird species classification (IEEE, 2016), pp. 396–401. https://doi.org/10.1109/ictai.2016.0067
39. L. Nanni, Y. M. G. Costa, D. R. Lucio, C. N. Silla Jr, S. Brahnam, Combining visual and acoustic features for audio classification tasks. Pattern Recogn. Lett. **88**, 49–56 (2017)
40. R. Nosaka, C. H. Suryanto, K. Fukui, in *Asian Conference on Computer Vision*. Rotation invariant co-occurrence among adjacent lbps (Springer, 2012), pp. 15–25. https://doi.org/10.1007/978-3-642-37410-4_2
41. T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. Pattern Anal. Mach. Intell. IEEE Trans. **24**(7), 971–987 (2002)
42. V. Ojansivu, J. Heikkilä, in *Image and Signal Processing*, ed. by A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass. Blur insensitive texture classification using local phase quantization (Springer, Berlin Heidelberg, 2008), pp. 236–243
43. S. Oramas, O. Nieto, F. Barbieri, X. Serra, Multi-label music genre classification from audio, text, and images using deep features. arXiv preprint (2017). arXiv:1707.04916
44. F. Pachet, A. Zils, in *ISMIR*. Automatic extraction of music descriptors from acoustic signals, (2004)
45. J. Pons, X. Serra, in *Acoustics, Speech and Signal Processing (ICASSP) 2017 IEEE International Conference on*. Designing efficient architectures for modeling temporal features with convolutional neural networks (IEEE, 2017), pp. 2472–2476. https://doi.org/10.1109/icassp.2017.7952601
46. J. Salamon, J. P. Bello, A. Farnsworth, S. Kelling, in *Acoustics, Speech and Signal Processing (ICASSP) 2017 IEEE International Conference on*. Fusing shallow and deep learning for bioacoustic bird species classification (IEEE, 2017), pp. 141–145. https://doi.org/10.1109/icassp.2017.7952134
47. M. San Biagio, M. Crocco, M. Cristani, S. Martelli, V. Murino, in *Computer Vision (ICCV) 2013 IEEE International Conference on*. Heterogeneous auto-similarities of characteristics (hasc): exploiting relational information for classification (IEEE, 2013), pp. 809–816. https://doi.org/10.1109/iccv.2013.105
48. J. Schlüter, S. Böck, in *6th International Workshop on Machine Learning and Music (MML)*. Musical onset detection with convolutional neural networks, (Prague, Czech Republic, 2013)
49. M. R. Schroeder, B. S. Atal, J. Hall, Optimizing digital speech coders by exploiting masking properties of the human ear. J. Acoust. Soc. Am. **66**(6), 1647–1652 (1979)
50. L. Sifre, S. Mallat, in *ESANN, vol. 44*. Combined scattering for rotation invariant texture analysis, (2012), pp. 68–81
51. S. Sigtia, S. Dixon, in *Acoustics, Speech and Signal Processing (ICASSP) 2014 IEEE International Conference on*. Improved music feature learning with deep neural networks (IEEE, 2014), pp. 6959–6963. https://doi.org/10.1109/icassp.2014.6854949
52. C. N. Silla Jr, A. L. Koerich, C. A. A. Kaestner, in *ISMIR*. The latin music database, (2008), pp. 451–456
53. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint (2014). arXiv:1409.1556
54. T. Song, H. Li, F. Meng, Q. Wu, J. Cai, Letrist: locally encoded transform feature histogram for rotation-invariant texture classification. IEEE Trans. Circ. Syst. Video Technol. (2017). https://doi.org/10.1109/tcsvt.2017.2671899
55. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Rethinking the inception architecture for computer vision, (2016), pp. 2818–2826. https://doi.org/10.1109/cvpr.2016.308
56. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Going deeper with convolutions, (2015), pp. 1–9. https://doi.org/10.1109/CVPR.2015.7298594
57. G. Tzanetakis, P. Cook, Musical genre classification of audio signals. IEEE Trans. Speech Audio Process. **10**(5), 293–302 (2002)
58. C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, *The Caltech-UCSD Birds-200-2011 Dataset. Tech. Rep. CNS-TR-2011-001*. (California Institute of Technology, 2011)
59. C. Y. Wang, A. Santoso, S. Mathulaprangsan, C. C. Chiang, C. H. Wu, J. C. Wang, in *Multimedia and Expo (ICME) 2017 IEEE International Conference on*. Recognition and retrieval of sound events using sparse coding convolutional neural network (IEEE, 2017), pp. 589–594. https://doi.org/10.1109/icme.2017.8019552
60. Q. Wang, P. Li, L. Zhang, W. Zuo, Towards effective codebookless model for image classification. Pattern Recogn. **59**, 63–71 (2016)
61. J. Xie, M. Zhu, Handcrafted features and late fusion with deep learning for bird sound classification. Ecol. Informa. **52**, 74–81 (2019)
62. Y. Yovel, M. O. Franz, P. Stilz, H. U. Schnitzler, Plant classification from bat-like echolocation signals. PLoS Comput. Biol. **4**(3), e1000,032 (2008)
63. G. Zhao, T. Ahonen, J. Matas, M. Pietikainen, Rotation-invariant image and video description with local binary pattern features. IEEE Trans. Image Process. **21**(4), 1465–1477 (2012)
64. Z. Zhao, S. h. Zhang, Z. y. Xu, K. Bellisario, N. h. Dai, H. Omrani, B. C. Pijanowski, Automated bird acoustic event detection and robust species classification. Ecol. Informa. **39**, 99–108 (2017)
65. Z. Zhu, X. You, C. P. Chen, D. Tao, W. Ou, X. Jiang, J. Zou, An adaptive hybrid pattern for noise-robust texture analysis. Pattern Recogn. **48**(8), 2592–2608 (2015)

Publisher's Note