Recurrent Convolutional Neural Networks for Large Scale Bird Species Classification

Gaurav Gupta (ggaurav@usc.edu)
University of Southern California
Meghana Kshirsagar
Microsoft (United States)
Ming Zhong
Microsoft (United States)
Shahrzad Gholami
Microsoft (United States)
Juan Lavista Ferres
Microsoft (United States)

Research Article

Keywords: large-scale prediction, bird acoustics, Cornell Bird Challenge (CBC) dataset, Convolutional Neural Network (CNN)

Posted Date: March 11th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-275942/v1

Recurrent Convolutional Neural Networks for large scale Bird species classification

Gaurav Gupta[1,*], **Meghana Kshirsagar**[2], **Ming Zhong**[2], **Shahrzad Gholami**[2], **and Juan Lavista Ferres**[2]

[1]Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA 90089, USA.
[2]AI for Good Research Lab, Microsoft, Redmond, WA 98052, USA.
[*]ggaurav@usc.edu

ABSTRACT

We present a deep learning approach towards the large-scale prediction and analysis ofr bird acoustics from 100 different bird species. We use spectrograms constructed on bird audio recordings from the Cornell Bird Challenge (CBC) dataset, which includes recordings with background noise, of multiple and potentially overlapping bird vocalizations per audio. Our experiments show that a hybrid modeling approach that involves a Convolutional Neural Network (CNN) for learning the representation for a slice of the spectrogram and a Recurrent Neural Network (RNN) for the temporal component to combine across time-points leads to the most accurate model on this dataset. We show results on a spectrum of models ranging from stand-alone CNNs to hybrid models of various types obtained by combining CNNs with CNNs or RNNs of the following types: Long Short-Term Memory (LSTM) networks, Gated Recurrent Units (GRU) and Legendre Memory Units (LMU). The best performing model achieves an average accuracy of 67% over the 100 different bird species, with the highest accuracy of 90% for the Red crossbill. We further analyze the learned representations visually and find them to be intuitive, where we find that related bird species are clustered close together. We present a novel way to empirically interpret the representations learned by the LMU-based hybrid model which shows how memory channel patterns over time change with spectrograms.

1 Introduction

Recent reports of shrinking bird populations world-wide[1,2] have emphasized the importance of monitoring of wild bird populations and protecting biodiversity. With this increasing need, automated audio recorders enable systematic recordings of environmental sounds and have recently opened new opportunities for ecological research and conservation practices. As many bird species have high vocal activities, bioacoustics has become one of the ideal ways to study them. Passive acoustic monitoring (PAM) of biological sounds can provide long-term and standardized data of the composition and dynamics of animal communities. Many bird species produce clear and consistent sounds, thus making acoustic surveys a reliable method to estimate the abundance, density, and occupancy of species[3,4]. Further, visual monitoring is difficult for many small and elusive birds, for cryptic species[5], and for species found in ecosystems difficult to reach for ecologists[6]. Besides, acoustic monitoring of birds is also helpful for other conservation activities, such as measuring forest restoration[7], and studying the impact of wild fires[8].

With the increasing volume of available audio recordings and the development of machine learning algorithms, autonomous classification of animal sounds has recently attracted a wide range of interests. Before deep learning gained wide-spread popularity, prior work had focused on the feature extraction from raw audio recordings and followed by some classification models, such as Hidden Markov Model[9,10], Random Forest[11], and Support Vector Machines[12]. While these methods demonstrated the successful use of machine learning approaches, their major limitation has been that most of the features need to be manually identified by a domain expert in order to make patterns more visible for the learning algorithms to work. In comparison, deep learning algorithms try to learn high-level features from the data in an incremental manner, which eliminates the need for domain expertise and hard core feature extraction efforts. Deep learning networks do not require human intervention, as multiple layers in neural networks place data in a hierarchy of different concepts, which ultimately learn from their own mistakes.

The use of deep learning for sound detection has spanned multiple domains, ranging from music classification to animal classification/detection (for example, marine species, avian, etc.). Among the related call detection and species classification works in the bioacoustics field, most of them adopted the methodology of using Convolutional Neural Networks (CNN) to classify the spectrograms or mel-spectrograms extracted from raw audio clips. These works achieved great success and the

deep learning models performed well with high classification accuracy to detect the presence or absence of calls from a particular species, or to classify calls from multiple species. While this method works well by transforming the raw audio into a spectrogram and then treating it as an image classification task, it does not take into consideration of the underlying temporal dependence characteristics of the species calls. It is worth noting that, different from the images with real objects, the x- and y-axis of spectrograms have specific implications (i.e., time and frequency, respectively, see Figure 1), and the time component embedded in the acoustics data shall contain important information for the corresponding classification tasks. Besides, some commonly used data augmentation techniques for image classification, such as rotation and flipping, may not make intuitive sense when applying to spectrograms generated from the acoustics data.

Our work makes the following contributions: (1) we propose a hybrid deep learning model that incorporates the benefit of convolutional and recurrent neural network models, capturing both spatial and temporal dependence of the bioacoustics data (2) our models achieve a better performance than previous ImageNet-based models that have been popular in prior work (3) our models have 7 times fewer parameters than stand-alone convolutional neural networks such as VGG16 (4) we present a novel empirical way to interpret the memory channels of the temporal component of our model (5) we present a way for ecologists to visualize the learned representations on different bird species.

2 Results

2.1 Dataset

The bird call classification 'Cornell Bird Challenge' (CBC) dataset[13] along with its extension, is used which consists of a total of 264 bird species with around 9 to 1778 audio samples per species. For the challenge, CBC obtained the data from xeno-canto.org. The raw audio samples vary in length from 5 seconds to 2 minutes. Since some classes have very few samples, we take 100 classes of birds by picking classes that had the highest numbers of samples and ensuring that each class has at least 100 samples and are close to balanced. Due to the variable length of the audio samples, we used a fixed-length: the first 7 sec of each audio clip as input and ignore audios that are shorter, resulting in a total of 15,032 samples across the 100 classes. We settled on the heuristic of taking the first 7 sec based on the criterion used for data curation by xeno-canto.org which requests bird audio contributors to trim the non-focal sounds and ensure that the specific bird species (focal sound) is heard within the first few seconds of the audio. For the purpose of training machine learning models, we split the dataset into 80% training, 10% validation, and 10% test examples. The raw audio clips are transformed to a mel-spectrogram based representation (see Figure 1 and Methods) using the librosa[14] package.

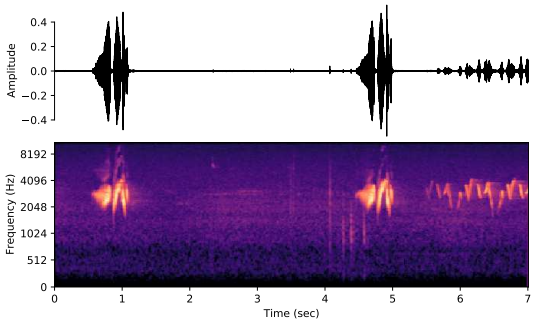


Figure 1. Audio spectrogram representation: The raw audio signal is transformed using the Fourier transform into a mel-spectrogram image. The frequency on the y-axis is in the mel scale.

2.2 Comparing Models

We train several variants of hybrid models and compare their average test accuracy using 5-fold cross-validation to that of baseline models. Specifically, we compare (i) the ImageNet models VGG16[15], ResNets[16] trained on a single spectrogram of the entire audio clip which we term as 'stand-alone' models. Next, (ii) hybrid models with window slides of the raw audio, and then spectrogram of each slide as an input using convolutional neural network (CNN) for representation and either CNN or recurrent neural network (RNN) for temporal correlation (see Methods). In Table 1 we show the test accuracy for stand-alone models as well as hybrid models. For the definitions of CNN and TCNN see section Methods. The ImageNet based models (stand-alone) lag behind the hybrid model in test accuracy which shows that explicitly using the temporal component in the models helps bird sound classification. We can make the following conclusions from the results in Table 1: (a) as we increase the complexity of the CNN from CNN1 to CNN3 (going downwards in the table), we see better test accuracy for all the hybrid models. (b) increasing the size of TCNN does not necessarily increase the test accuracy. (c) increasing the size of the hidden state in each RNN (going from 128 to 512) increases the test accuracy for all RNNs (d) however, increasing the number of layers in the RNN does not necessarily improve the performance. We refer the reader to Supplementary Table 1,2 for the complete results. For most of the models, one or two layers result in the best performance across all RNNs. Overall, the temporal block with the Gated Recurrent Unit (GRU) units achieves the best accuracy, while using GRU and Legendre Memory Units (LMU) together also gives a similar accuracy to the best model but with less trainable parameters. We discuss the aspect of trainable parameters for each model later in this section.

In Table 1 we compared the test accuracy of the models, which gives us information about the prediction, i.e the maximum value of the softmax outputs. Now, we compare the softmax distribution of the models in Figure 2 in the following manner. First, for each trained model, the softmax outputs of all the test samples are concatenated. Second, the concatenated softmax

stand-alone models													
		ResNet18 0.516				ResNet50 0.537					VGG16 0.619		
Temporal correlation with CNN/RNN													
Size	TCNN	LSTM			GRU			LMU			GRU+LMU		
		Layers											
		1	2	3	1	2	3	1	2	3	1	2	3
		CNN1											
S	0.49	0.57	0.56	0.55	0.60	0.58	0.58	0.57	0.55	0.54	–	0.57	0.56
L	0.50	0.62	0.61	0.61	0.63	0.64	0.63	0.61	0.61	0.59	–	0.63	0.63
		CNN2											
S	0.56	0.60	0.58	0.55	0.62	0.60	0.60	0.58	0.557	0.56	–	0.59	0.59
L	0.55	0.62	0.61	0.60	0.63	0.63	0.64	0.63	0.62	0.60	–	0.63	0.63
		CNN3											
S	0.58	0.66	0.62	0.58	0.64	0.64	0.62	0.65	0.64	0.63	–	0.63	0.62
L	0.61	0.66	0.63	0.64	0.66	**0.67**	0.65	0.65	0.65	0.64	–	**0.66**	0.65

Table 1. Test accuracy comparison on the CBC dataset: **Top**: Models without any explicit temporal layer. The input is a single spectrogram for sound sample. **Bottom**: A comprehensive comparison of models test accuracy using CNN/RNN for temporal correlation. The complexity of CNN used for representation increase from top to bottom. The best accuracy achieved is shown in bold. For each representation CNN*, a small (S) and a large (L) temporal layer is presented. For RNNs the S/L refer to hidden size of 128/512, while for TCNN S/L refers to TCNN1/TCNN3.

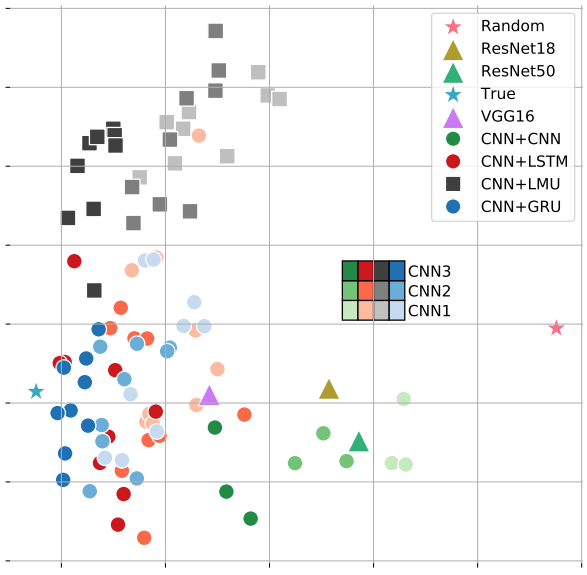


Figure 2. Comparison of models. A PCA plot for test outputs ($\in \mathbb{R}^{100}$) of various models. The *Perfect* point denotes correct 100×1 one-hot test label output, while *Random* denotes uniform probability (or maximum entropy) 100×1 output.

stand-alone models						
		ResNet18 14		ResNet50 24		VGG16 134
Temporal correlation with CNN/RNN						
Size	TCNN	LSTM		GRU		LMU
		Layers				
		1	3	1	3	1	3
		CNN1					
S	3.8	1.2	1.4	1.1	1.3	1.0	1.1
L	9.8	2.8	7.0	2.4	5.1	1.6	2.8
		CNN2					
S	5.3	2.9	3.1	2.8	3.0	2.6	2.7
L	11.3	4.8	9.0	4.3	7.5	3.3	4.5
		CNN3					
S	17.6	15.4	15.7	15.3	15.5	15.0	15.1
L	23.6	18.2	22.4	17.4	20.5	15.9	17.0

Table 2. Model complexity: Total number of trainable parameters (in Millions) for different models.

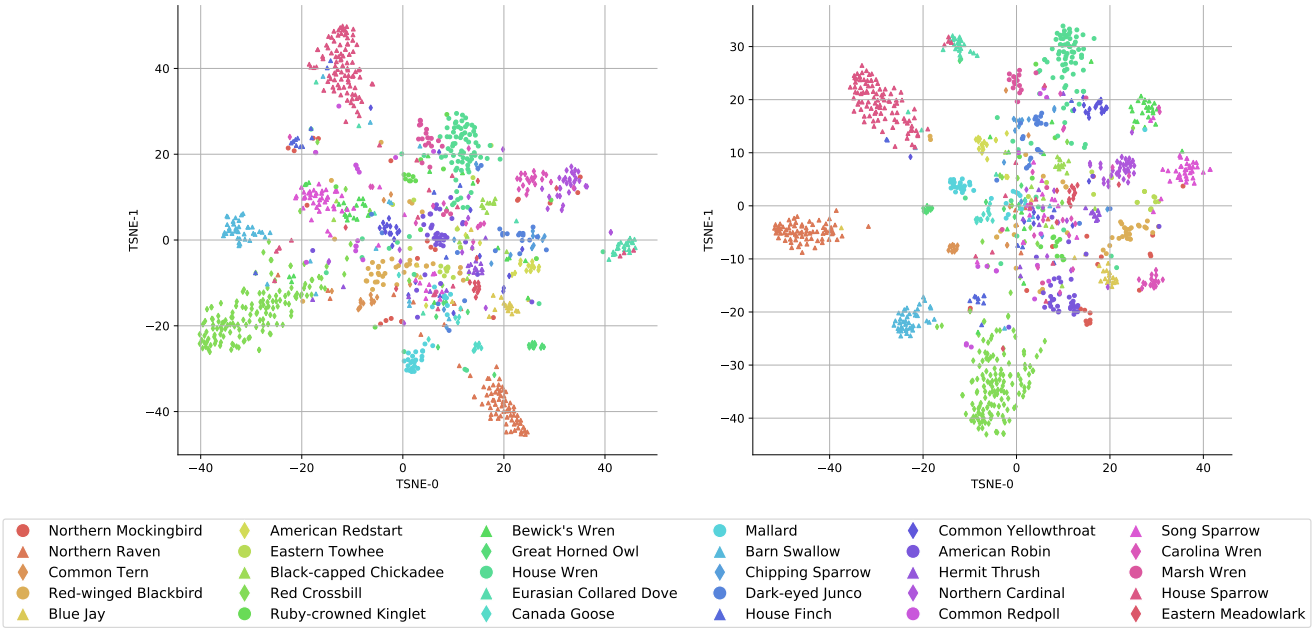


Figure 4. Samples model representation: t-SNE plot along two dimensions for 30 bird species with most number of samples. The test samples embedding is shown for CNN3+LMU in *left*, and CNN3+GRU in *right* with hidden size of 512 for each model.

vectors are then projected along the two dimensions with the maximum variance by performing Principal Component Analysis (PCA). We observe that the hybrid models with CNN for both representation and temporal components are clustered together with the stand-alone models, and different from the hybrid models that use RNNs for the temporal component. The hybrid models with RNNs that have gating mechanisms like Long Short-Term Memory networks (LSTM) and GRU are very close to each other in the PCA plot. The hybrid models with LMU are clustered together and are away from LSTM and GRU. For reference, we also show the two corner cases of (i) 'true', which is the actual one-hot label of the test samples, and (ii) 'random' which assigns equal probability to all the classes.

For different models, we also show the model complexity in terms of total trainable parameters in Table 2. We conclude that on the CBC dataset, the stand-alone ImageNet-based models with higher trainable parameters do not deliver higher test accuracy. The hybrid models offer dual advantages in terms of less model complexity as well as higher test classification accuracy. Next, we compare the class-wise prediction accuracy of the best stand-alone model (VGG16), and the best GRU, LMU model from Table 1 in Figure 3. We see that GRU, LMU has more number of classes in higher prediction accuracy bands as compared to VGG16.

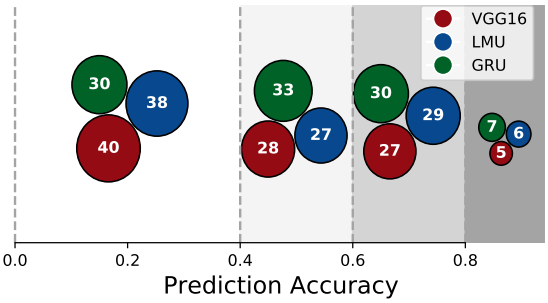


Figure 3. Model class-wise predictions: For VGG16, and the best GRU, LMU model (from Table 1), the percentage of classes that each model has prediction accuracy in the given shaded brackets.

2.3 Visualizing the learned representations

We now analyze the representations learned by the trained models for different bird species. For each audio sample, we obtain the representation by taking the output of the penultimate layer of the model, and in Figure 4 we show the t-SNE embeddings in two dimensions for 1522 test samples over 30 bird species. The 30 bird species with the most number of samples are picked from total of 100 species data. The embedding for two different models CNN3+(LMU, GRU) with a hidden size of 512 is shown in the left and right plots, respectively. For both models, we see that the bird species like Red Crossbill, Northern Raven and House Sparrow that have distinct calls appear in tight-knit clusters (for birds code see Supplementary Table 3, and for further related information we refer[17]). On the other hand, species like Northern Mockingbird which belong to the mimic-thrush family, *Mimidae*, have spread-out examples due to the heterogeneity of their calls. We find Northern Mockingbird examples in clusters belonging to several species of Wrens, the Blue Jay and American Robin. Further, House Wren and Marsh Wren examples are

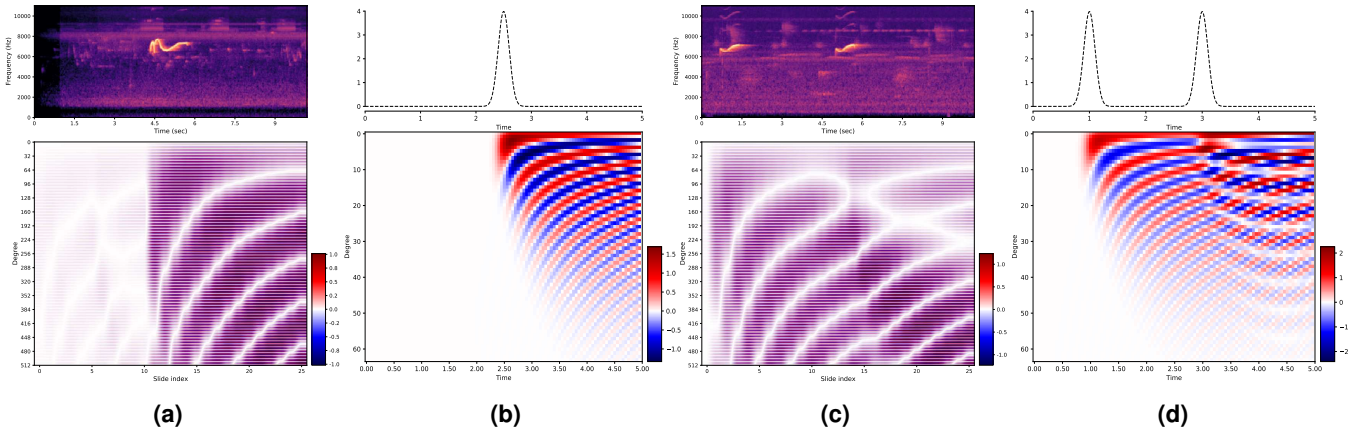


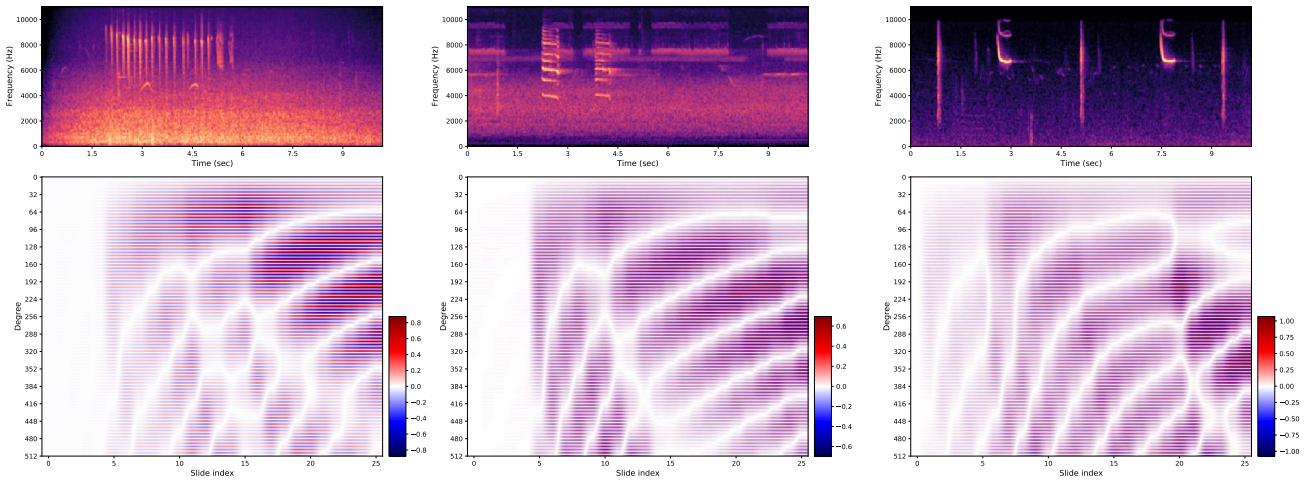
Figure 5. LMU memory channels: LMU memory channels behavior vs time for input signals in the form of pulse. For each subplot, the input is shown in the top and the bottom is memory channels value vs time. A bird sound test sample with single/double spectrogram pulse is shown in (a)/(c), respectively. The time in spectrogram is synchronized with the slide index in accordance with the chosen values of (W_s, H_s) (see Methods). Simulated version of single/double pulse input is shown in (b)/(c), respectively.

projected close together by both methods due to their similar calls, whereas Carolina Wren and Bewick's Wren are farther. Using the embedding plot we can further identify the clustered species and the species that are close to each other which could provide insights to the bird ecologists. The complete embedding plots with all the species is provided in the Supplementary materials.

2.4 Analyzing Memory

The deep learning models like the ones we have seen in the previous section deliver good performance. But understanding their mechanism i.e. interpreting what the models have learned, is still difficult. The gating mechanisms employed in LSTM and GRU are difficult to interpret w.r.t how they act upon different input signals like sounds. On the other hand, an RNN like LMU is based on an entirely different machinery that employs a state-space model and updates the memory channels using the dynamical equation (3) with matrices A, B in (3) constructed using Legendre polynomials. Another interpretation for the LMU memory mechanism which makes more sense is that: LMU memory equation (3) projects the entire input signal history into a fixed number of orthogonal Legendre polynomials[18] in an online fashion. The projection is made at each time-step, and to avoid the repeated computation of projections, the dynamical equation in (3) is used (see Methods). We demonstrate this projection behavior of the LMU in Figure 5. We see in Figure 5a that the trained LMU model starts to populate the memory channels upon the first arrival of the pulse in the spectrogram. For the later time points, the memory channel values are transformed to register the signal history. In Figure 5b, we demonstrate this behavior by simulating a pulse input and projecting the signal history at any time t onto 64 orthogonal Legendre polynomials but *without* using the dynamical equation (3). Before the t=2 sec time-point, the projections are zero as there is no signal history. We then see the patterns of memory channels (similar to Figure 5a) as the pulse arrives. A similar behavior is shown for the bird spectrogram with two pulses in Figure 5c and a simulated version of two pulses in Figure 5d. We see that the arrival of the second pulse changes the evolution pattern of the memory channels.

The LMU memory channel values with time are compared for three different bird species samples in the Figure 6. We see that, irrespective of the different bird species, the memory starts populating when the significant energy in the spectrograms is first detected. Some misalignment exists between the beginning of spectrogram pulses and the corresponding response in the memory channels due to the granularity of the chosen stride parameters (W_s, H_s) (see Methods for more details). We make the following two conclusions: (i) for a pulse-like behavior where the spectrogram has energy concentrated in a short-time duration, the memory channels have fading in a smooth fashion as we see in Figure 6(b). While for the spectrograms with energy spread out in time, we see more frequent changes in the memory channels with circular patterns in Figure 6(a). Next, (ii) compared to the double pulse example, as we see in Figure 5(c), where the spectrogram has energy in a narrow frequency range of 6-7 KHz, the case where energy is scattered in a wider range of 4-9 KHz in Figure 6(b) and 8-10 KHz in Figure 6(c) has different response for the memory channels.



(a) Anna's Hummingbird **(b)** Blue Jay **(c)** Red Winged Blackbird

Figure 6. LMU memory channels for real examples: Variations of LMU memory channel values with time for three different bird species spectrograms in (a)-(c). The time in spectrogram is synchronized with the slide index in accordance with the chosen values of (W_s, H_s) (see Methods).

3 Methods

3.1 Spectrograms

The frequency transformation of a time-domain signal using mel-spectrograms have been shown to be better than short time Fourier transform (STFT), mel-frequency cepstral coefficients (MFCCs)[19] in the works[20,21]. We compute mel-spectrogram using librosa[14] for the 7 sec clipped audio signals. The audio is re-sampled at 32KHz and a total of 128 mel filter banks were used. The Fast Fourier Transform (FFT) length is taken to be 2048, and the hop-length for computing spectrogram is taken as 512.

3.2 Models

Stand-alone: The ImageNet models, for example, VGG16, ResNet are used as classifier using spectrograms as the 2-dimensional input. The neurons in the final layer are selected as per the number of classes in the dataset. For CBC, since we are taking 100 classes, the output layer has 100 neurons.

Hybrid: The hybrid models use a sliding window mechanism for the input. The raw audio clip is traversed via a sliding window of length W_s and hop length H_s. Each hop of window results in a clipped audio of length W_s which is transformed to frequency domain using mel-spectrograms. The values of (W_s, H_s) used in this work are (500, 250) msec. For a 7-second audio clip, a total of 26 slides are made with the used values of W_s, H_s. After input, the hybrid models have three parts, (i) Representation, (ii) Temporal correlation, and (iii) Classification. The representation block uses a CNN to generate representative features from the input slides. After concatenating the representative feature vectors from multiple slides, the resulting 2-dimensional array is used as an input to the next Temporal correlation block. The schematic for hybrid models is shown in Figure 7. The output from the temporal correlation block is fed to the final classification block to produce the softmax outputs.

Representation Models: In this work, we use three CNN (CNN1, CNN2, and CNN3) of different lengths for the representation block. **(a)** The CNN1 has one block of [Convolution, MaxPool] with 32 3×3 filters. This is followed by a block of [Convolution, Convolution, MaxPool] of 64, 64 3×3 filters. Finally, we have two [Convolution, Convolution, Convolution, MaxPool] blocks of 128, 128, 128 3×3 filters. Every Convolution filter layer is followed by a Batch normalization layer and ReLU operation. The MaxPool is set to down sample with the factor of 2. **(b)** The CNN2 has one block of [Convolution, Convolution, MaxPool] with 32, 64 3×3 filters. This is followed by four blocks of [Convolution, Convolution, Convolution, MaxPool] with (64, 64, 64), (128, 128, 128), (128, 128, 128), (256, 256, 256) 3×3 filters. Every Convolution filter layer is followed by a Batch normalization layer and ReLU operation. The MaxPool is set to downsample with the factor of 2. Lastly, **(c)** the CNN3 has two blocks of [Convolution, Convolution, MaxPool] (64, 64), (128, 128) 3×3 filters. This is followed by three blocks of [Convolution, Convolution, Convolution, MaxPool] wth (256, 256, 256), (512, 512, 512), (512, 512, 512) 3×3 filters. Every Convolution filter layer is followed by a Batch normalization layer and ReLU operation. The MaxPool is set to downsample with the factor of 2.

Temporal Models: The temporal block either uses CNN (as shown in Figure 7A), or RNN (as shown in Figure 7B). In this

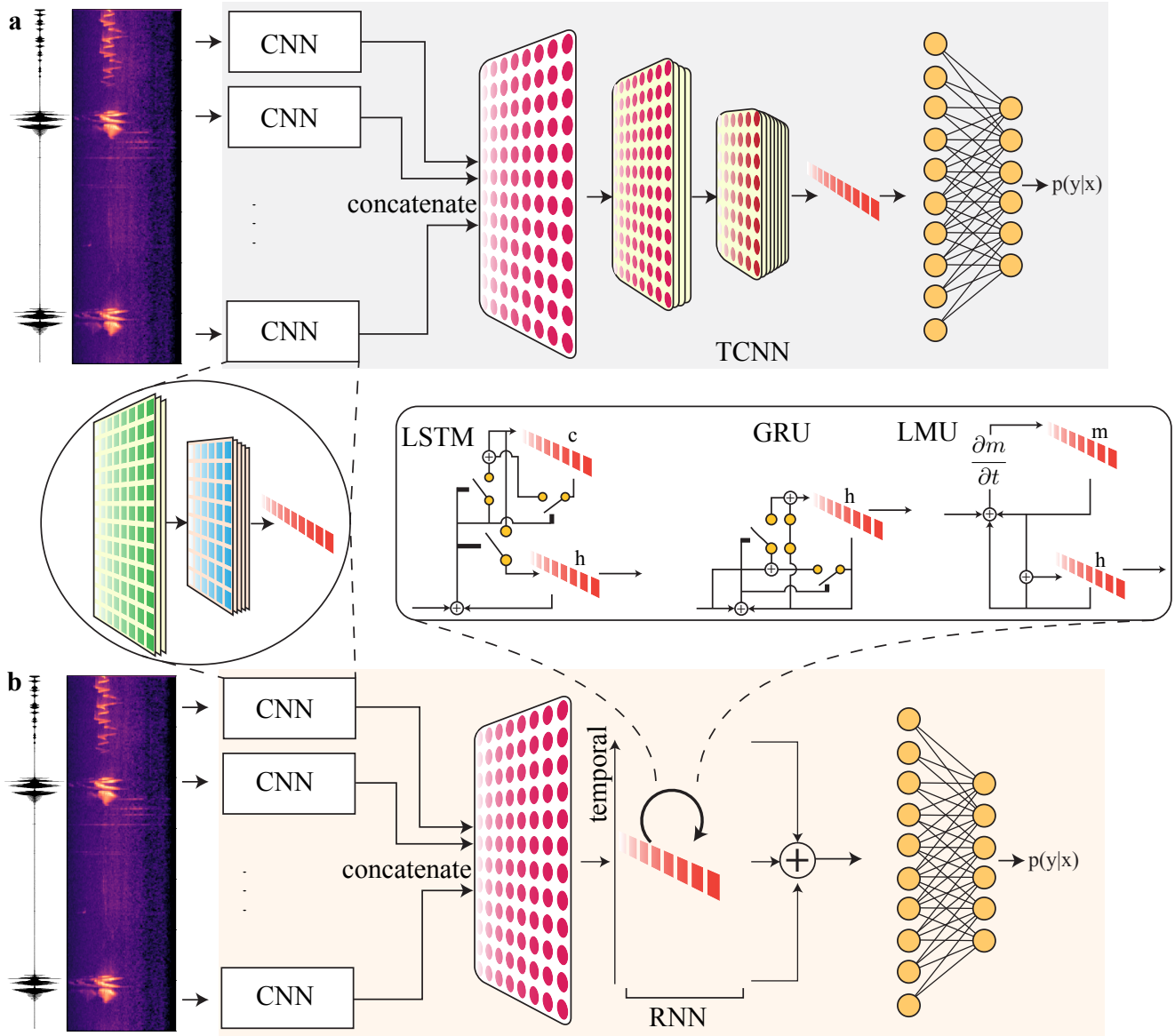


Figure 7. A schematic of hybrid models for classification. Model pipeline using CNN for representation and using another TCNN in **a**, RNN in **b** for temporal correlation extraction. The CNN outputs are concatenated before feeding to the temporal layer in **a**, **b**.

work, the models using CNN in the temporal block use three networks (TCNN1, TCNN2, and TCNN3) of different lengths. **(a)** The TCNN1 has one block of [Convolution, Convolution, MaxPool] with (64, 64) 3×3 filters. This is followed by three blocks of [Convolution, Convolution, Convolution, MaxPool] with (128, 128, 128), (128, 128, 128) and (256, 256, 256) with 3×3 filters. **(b)** The TCNN2 has one block of [Convolution, Convolution, MaxPool] with (64, 64) 3×3 filters. This is followed by three blocks of [Convolution, Convolution, Convolution, MaxPool] with (128, 128, 128), (256, 256, 256) and (256, 256, 256) with 3×3 filters. **(c)** The TCNN3 has one block of [Convolution, Convolution, MaxPool] with (64, 64) 3×3 filters. This is followed by three blocks of [Convolution, Convolution, Convolution, MaxPool] with (128, 128, 128), (256, 256, 256) and (512, 512, 512) with 3×3 filters. Every Convolution filter layer is followed by a Batch normalization layer and ReLU operation. The MaxPool is set to downsample with the factor of 2.

The hybrid models with temporal block using RNN has three variations in this work, namely LSTM, GRU, and LMU. The LSTM uses a hidden state h and also maintains a cell state c. The recursive update equations for the LSTM are shown in (1). The GRU has a compact gating mechanism compared to the LSTM and has two gates. The update equations for the GRU are stated in (2). The LMU uses a memory concept and updates the memory using projections onto Legendre polynomials. The

update equations (as shown in (3) are less expensive in terms of trainable parameters due to the fixed values of $\bar{A}, \bar{B}$ matrices. We refer the reader to original work[22] for more details.

Finally, the output of the temporal block is used as an input to the Classification block which implements fully-connected multi-layer perceptron (MLP). The classification block has one layer of 512 neurons with ReLu non-linearity followed by dropout (with probability 0.5) and output layer of neurons according to the class size of dataset. In the case of temporal block being RNN, the outputs at all time-steps are summed before feeding to the classification block.

$$
\begin{aligned}
f_t &= \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f),\\
i_t &= \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i),\\
o_t &= \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o),\\
\tilde{c}_t &= \sigma(W_{cx}x_t + W_{ch}h_{t-1} + b_c),\\
c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t,\\
h_t &= o_t \odot \tanh(c_t).
\end{aligned}
\tag{1}
$$

$$
\begin{aligned}
z_t &= \sigma(W_{zx}x_t + W_{zh}h_{t-1} + b_z),\\
r_t &= \sigma(W_{rx}x_t + W_{rh}h_{t-1} + b_r),\\
\tilde{h}_t &= \tanh(W_{ch}(r_t \odot h_{t-1}) + W_{cx}x_t + b_h),\\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t.
\end{aligned}
\tag{2}
$$

$$
\begin{aligned}
h_t &= \tanh(W_x x_t + W_h h_{t-1} + W_m m_t),\\
u_t &= e_x^T x_t + e_h^T h_{t-1} + e_m^T m_t,\\
m_t &= \bar{A} m_{t-1} + \bar{B} u_t.
\end{aligned}
\tag{3}
$$

3.3 Analyzing Memory

The other interpretation for LMU mechanism, apart from the state-space representation, is projecting the memory onto a fixed set of orthogonal basis. Hence, the LMU works by the repeated projection of the entire history of hidden states h_t and the input x_t, $t \geq 0$ onto a fixed number of Legendre polynomials. The Legendre polynomials are a class of orthogonal polynomials with the following property.

$$
\int_{-1}^{1} P_m(x)P_n(x)dx = \begin{cases} 0, & m \neq n \\ \frac{2}{2n+1}, & m = n \end{cases}, \tag{4}
$$

where $P_m(x)$ is the Legendre polynomial with degree m. The Legendre polynomials also satisfy the following

$$
P'_{m+1} = (m+1)P_m + xP'_m, \tag{5}
$$

$$
(2m+1)P_m = P'_{m+1} - P'_{m-1}, \tag{6}
$$

$$
P_m(1) = 1, \quad P_m(-1) = (-1)^m. \tag{7}
$$

For a signal $f(t)$, its projection along the mth degree Legendre polynomial is defined as

$$
c_m(t) = \int_0^t f(x)P_m(x)dx. \tag{8}
$$

In Figure 5(b),(c) we use (8) to show the projection coefficient variations over time with the maximum degree of 64. Directly evaluating the projections at each time-step t using (8) is not computationally feasible, especially when the time-horizon is large. However, due to the recurrence properties of the Legendre polynomials (5),(6) a dynamical equation like (3) can be constructed to update the projection coefficients recursively.

4 Related Work

During the past decade, deep convolutional neural network (CNN) architectures have demonstrated great potential in classification problems as well as other tasks, such as object detection and image segmentation. Some well-known CNN architectures include VGG16[15], ResNet[16], and DenseNet[23], among others. These models can successfully extract complex features from the images and differentiate a high number of potentially similar classes, and have recently gathered popularity in the field of bioacoustics as well. For example, there are some works using CNN, either based on the well-known architectures or customized architectures, to detect and classify the presence of whale acoustics[24,25], or classify calls from different bird species[26,27].

While CNN models usually include millions of parameters, training such a model typically requires a sufficiently large amount of data in order to achieve good performance. However, it is a time-consuming and expensive endeavor to obtain a manually labeled dataset in bioacoustics, and it may also be very challenging to collect enough labeled data in practice, especially if a species rarely calls or if a species is rare. Given this scenario, some bioacoustics research works used other

techniques in addition to CNN, including transfer learning with fine-tuning[28–30], pseudo-labeling[31], and using few-shot learning approaches[32].

Existing literature in recurrent and convolutional neural networks has extensively explored the classification task on the sequence and time-series datasets. While not explicitly modeling the temporal dependencies, fully convolutional networks, and ResNet architectures are shown to perform well for time-series classification in[33]. Vanilla recurrent neural nets were designed to capture temporal dependencies for sequence data[34,35]. However, they suffer from vanishing/exploding gradients[36]. As a remedy, more sophisticated recurrent neural net units that implement a gating mechanism, such as a long short-term memory (LSTM) unit[37] and gated recurrent unit (GRU)[38] are proposed in the literature. For the audio classification task, a gated Residual Networks model that integrates ResNet with a gate mechanism was shown to be promising[39]. To efficiently handle the temporal dependencies, the Legendre Memory Unit (LMU) was proposed as a novel memory cell for recurrent neural networks with theoretical guarantees for learning long-range dependencies[22,40]. It dynamically maintains information across long windows of time using relatively few resources via orthogonalizing its continuous-time history.

Hybrid models leverage the strengths of both convolutional and recurrent neural networks for learning from temporal or sequence data. They use convolutional layers to extract local patterns at each time-point and then couple the learned representations over multiple time-points using a recurrent component. As compared to the models that use another CNN layer to aggregate the representations across time-steps, the use of a recurrent structure allows them to better capture long-term dependencies in the input. Various choices of recurrent components have been tried, such as LSTMs, GRUs. A one-dimensional CNN coupled with a GRU was proposed in[41],[42] use an LSTM coupled to a CNN for audio classification,[43] develop a recurrent structure that is based on GRUs, with temporal skip connections to extend the temporal span of the information flow for modeling multi-dimensional time-series. A variety of CNN and RNN models are explored in[44] where superior performance of deep nets compared to some traditional machine learning models is demonstrated for automatic detection of endangered mammals species based on spectrograms. Hybrid models have shown improvements in accuracy over the baseline CNN-only models on various sound detection tasks in the recent literature[45,46]. Further, for the task of music tagging, Choi et al.[47] show that their convolutional recurrent neural network (CRNN), that also involves a GRU, does better in terms of training time and the number of parameters compared to the purely CNN-based prior architectures.

5 Conclusion

We have presented a comprehensive study of the deep learning models on a large bird acoustics dataset Cornell Bird Challenge (CBC). The deep learning models offer high prediction capability and at the same time lead to a design of a more automated pipeline. Although the Imagenet models are successful on the image classification and are also applied for the sound classification through spectrograms, they lack the temporal component. We found out that for sound dataset (CBC) hybrid models with an explicit temporal layer help. The hybrid models compared to the Imagenet models offered two-fold advantage of reduced model size as well as higher test accuracy. We also found out that larger models do not always result in the better test accuracy. In the context of RNN, in most cases, one or two layers were sufficient and resulted in more accurate models. In addition to the gating mechanisms based RNNs like Long-Short term memory (LSTM), and Gated recurrent units (GRU), we also present a novel hybrid model utilizing Legendre memory units (LMU). The LMU works on a different mechanism of orthogonalizing memory and offers the further advantage of long-range dependence as well as reduced model parameters. We have presented an empirical analysis of how LMU memory channels behave with time for different spectrogram inputs.

We have also analyzed how models are representing different bird species sound samples through the embedding plot. We found out that the birds with distinct calls (for example, Red crossbill, Northern raven, etc.) are packed together and are distant from each other. Some bird species with assorted calls are spread across other species representations.

The hybrid models with a built-in temporal layer have an additional requirement of a longer time sequence. For shorter time-series, learning dependencies across time components was found out to be difficult through RNNs. We have also found out that adding the attention mechanisms to the hybrid models with RNN does not help with the CBC dataset. Part of the reason could be that the bird call location in the input audio is very uncertain, even in the clipped version. In future work, we would extend the current models to detect multiple species of bird calls, and also applying the same analysis to different sound datasets, for example, marine animals detection.

References

1. Rosenberg, K. V. *et al.* Decline of the north american avifauna. *Science* **366**, 120–124 (2019).
2. Inger, R. *et al.* Common european birds are declining rapidly while less abundant species' numbers are rising. *Ecol. letters* **18**, 28–36 (2015).
3. Leach, E. C., Burwell, C. J., Ashton, L. A., Jones, D. N. & Kitching, R. L. Comparison of point counts and automated acoustic monitoring: detecting birds in a rainforest biodiversity survey. *Emu* **116**, 305–309 (2016).

4. Drake, K. L., Frey, M., Hogan, D. & Hedley, R. Using digital recordings and sonogram analysis to obtain counts of yellow rails. *Wildl. Soc. Bull.* **40**, 346–354 (2016).
5. Lambert, K. T. & McDonald, P. G. A low-cost, yet simple and highly repeatable system for acoustically surveying cryptic species. *Austral Ecol.* **39**, 779–785 (2014).
6. Burnett, K. *Distribution, abundance, and acoustic characteristics of Kohala forest birds*. Ph.D. thesis, University of Hawaii at Hilo (2020).
7. Owen, K. *et al.* Bioacoustic analyses reveal that bird communities recover with forest succession in tropical dry forests. *Avian Conserv. Ecol.* **15** (2020).
8. Furnas, B. J., Landers, R. H. & Bowie, R. C. Wildfires and mass effects of dispersal disrupt the local uniformity of type i songs of hermit warblers in california. *The Auk* **137**, ukaa031 (2020).
9. Aide, T. M. *et al.* Real-time bioacoustics monitoring and automated species identification. *PeerJ* **1** (2013).
10. Potamitis, I., Ntalampiras, S., Jahn, O. & Riede, K. Automatic bird sound detection in long real-field recordings: applications and tools. *Appl. Acoust.* **80**, 1–9 (2014).
11. Stowell, D. & Plumbley, M. D. Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning. *PeerJ* **2** (2014).
12. Tachibana, R. O., Oosugi, N. & Okanoya, K. Semi-automatic classification of birdsong elements using a linear support vector machine. *PLoS One* **9** (2014).
13. [Dataset] Cornell Lab of Ornithology. Cornell birdcall identification. https://www.kaggle.com/c/birdsong-recognition. Accessed: 06/15/2020.
14. McFee, B. *et al.* librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, vol. 8 (2015).
15. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *ICLR* (2015).
16. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *CVPR*, 770–778 (2016).
17. Billerman, S. M., Keeney, B. K., Rodewald, P. G. & Schulenberg, T. S. (eds.) *Birds of the World* (Cornell Laboratory of Ornithology, Ithaca, NY, USA, 2020). https://birdsoftheworld.org/bow/home.
18. Gu, A., Dao, T., Ermon, S., Rudra, A. & Re, C. Hippo: Recurrent memory with optimal polynomial projections (2020). 2008.07669.
19. Molau, S., Pitz, M., Schluter, R. & Ney, H. Computing mel-frequency cepstral coefficients on the power spectrum. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, vol. 1, 73–76 vol.1, DOI: 10.1109/ICASSP.2001.940770 (2001).
20. Choi, K., Fazekas, G. & Sandler, M. Automatic tagging using deep convolutional neural networks (2016). 1606.00298.
21. Dieleman, S. & Schrauwen, B. End-to-end learning for music audio. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6964–6968 (2014).
22. Voelker, A., Kajic, I. & Eliasmith, C. Legendre memory units: Continuous-time representation in recurrent neural networks. In *NeurIPS* (2019).
23. Huang, G., Liu, Z., van der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *CVPR*, 4700–4708 (2017).
24. Doriana, C., Leforta, R., Bonnela, J., Zaraderb, J.-L. & Adam, O. Bi-class classification of humpback whale sound units against complex background noise with deep convolution neural network (2017). 1702.02741.
25. Bergler, C. *et al.* Orca-spot: An automatic killer whale sound detection toolkit using deep learning. *Sci. Reports* **9** (2019).
26. Salamon, J., Bello, J. P., Farnsworth, A. & Kelling, S. Fusing shallow and deep learning for bioacoustic bird species classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2017).
27. Narasimhan, R., Fern, X. Z. & Raich, R. Simultaneous segmentation and classification of bird song using cnn. In *Proc. Int. Conf. Acoust. Speech, Signal Process*, 146–150 (2017).
28. Zhang, L., Wang, D., Bao, C., Wang, Y. & andKele Xu. Large-scale whale-call classification by transfer learning on multi-scale waveforms and time-frequency features. *Appl. Sci.* **9** (2019).
29. Berman, P. C., Bronstein, M. M., Wood, R. J., Gero, S. & Gruber, D. F. Deep machine learning techniques for the detection and classification of sperm whale bioacoustics. *Sci. Reports* **9** (2019).

30. Zhong, M. *et al.* Improving passive acoustic monitoring applications to the endangered cook inlet beluga whale. *The J. Acoust. Soc. Am.* **146** (2019).

31. Zhong, M. *et al.* Multispecies bioacoustic classification using transfer learning of deep convolutional neural networks with pseudo-labeling. *Appl. Acoust.* **166** (2020).

32. Thakura, A., Thapar, D., Rajan, P. & Nigam, A. Deep metric learning for bioacoustic classification: Overcoming training data scarcity using dynamic triplet loss. *The J. Acoust. Soc. Am.* **146** (2019).

33. Wang, Z., Yan, W. & Oates, T. Time series classification from scratch with deep neural networks: A strong baseline (2016). 1611.06455.

34. Williams, R. J. & Zipser, D. A learning algorithm for continually running fully recurrent neural networks. *Neural computation* **1**, 270–280 (1989).

35. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *nature* **323**, 533–536 (1986).

36. Bengio, Y., Simard, P. & Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* **5**, 157–166 (1994).

37. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural computation* **9**, 1735–1780 (1997).

38. Cho, K., Van Merriënboer, B., Bahdanau, D. & Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259* (2014).

39. Zeng, Y., Mao, H., Peng, D. & Yi, Z. Spectrogram based multi-task audio classification. *Multimed. Tools Appl.* **78**, 3705–3722 (2019).

40. Voelker, A. R. & Eliasmith, C. Improving spiking dynamical networks: Accurate delays, higher-order synapses, and time cells. *Neural Comput.* **30**, 569–609 (2018).

41. Xu, Y., Kong, Q., Huang, Q., Wang, W. & Plumbley, M. D. Convolutional gated recurrent neural network incorporating spatial features for audio tagging (2017). 1702.07787.

42. Keren, G. & Schuller, B. Convolutional rnn: an enhanced model for extracting features from sequential data (2016). 1602.05875.

43. Lai, G., Chang, W.-C., Yang, Y. & Liu, H. Modeling long- and short-term temporal patterns with deep neural networks (2017). 1703.07015.

44. Shiu, Y. *et al.* Deep neural networks for automated detection of marine mammal species. *Sci. Reports* **10**, 607 (2020).

45. Espi, M., Fujimoto, M., Kubo, Y. & Nakatani, T. Spectrogram patch based acoustic event detection and classification in speech overlapping conditions. In *2014 4th Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, 117–121 (2014).

46. Feng, L., Liu, S. & Yao, J. Music genre classification with paralleling recurrent convolutional neural network (2017). 1712.08370.

47. Choi, K., Fazekas, G., Sandler, M. & Cho, K. Convolutional recurrent neural networks for music classification (2016). 1609.04243.

Author contributions statement

G.G conducted the experiments, G.G., M.K., M.Z., and S.G designed the experiments and analysed the results. All authors reviewed the manuscript.

Competing interests

The author(s) declare no competing interests.

Figures

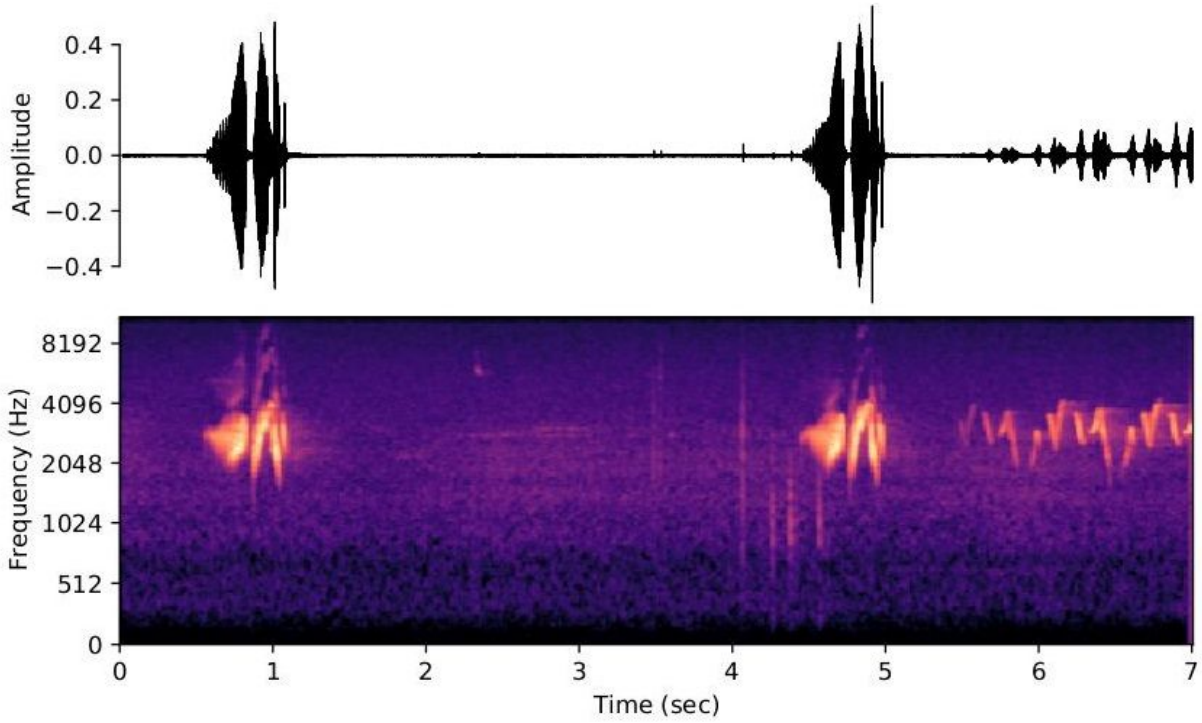


Figure 1

Audiospectrogramrepresentation: The raw audio signal is transformed using the Fourier transform into a mel-spectrogram image. The frequency on the y-axis is in the mel scale.

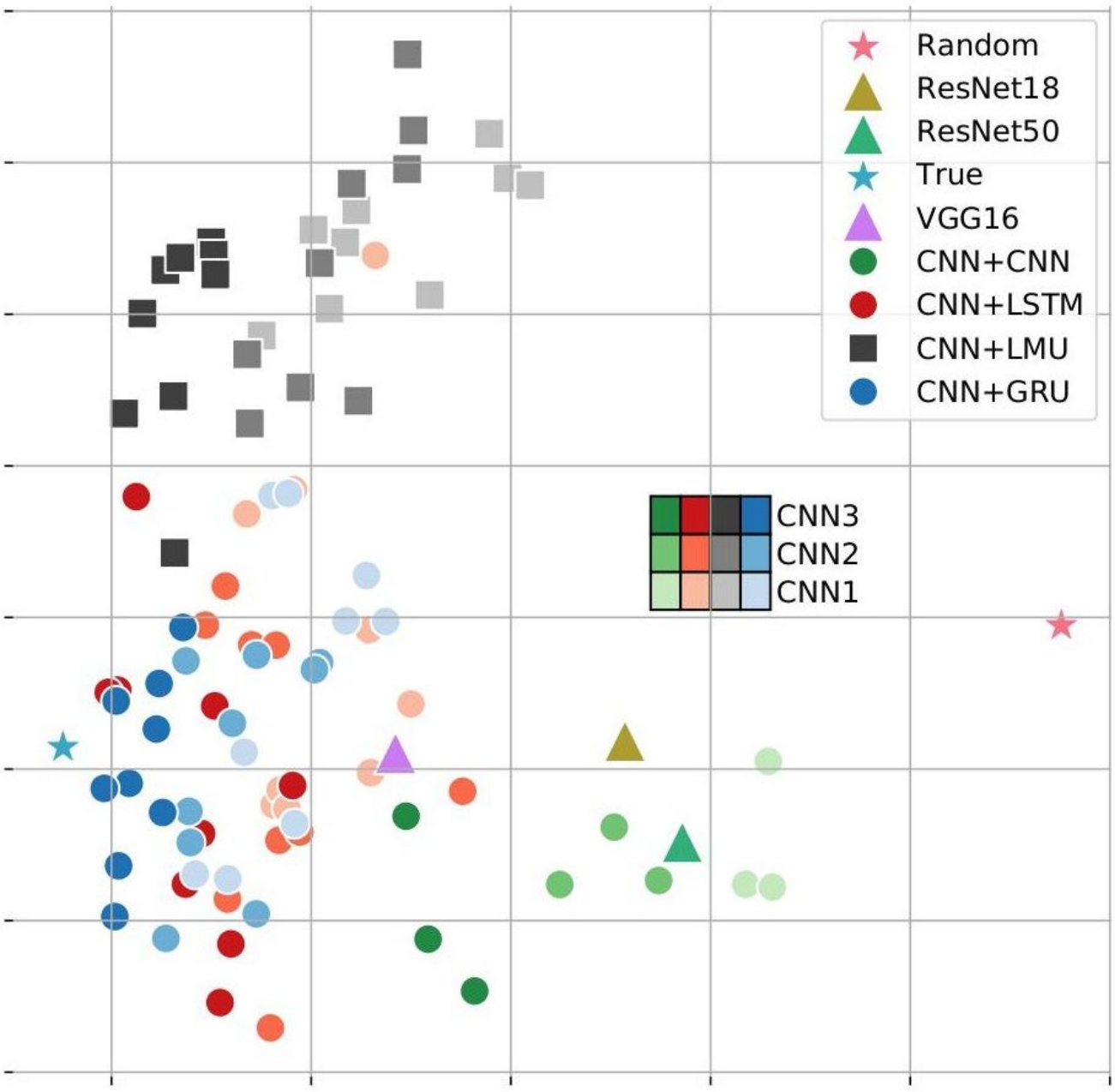


Figure 2

Comparisonofmodels. A PCA plot for test outputs (R100) of various models. The Perfect point denotes correct 100×1 one-hot test label output, while Random denotes uniform probability (or maximum entropy) 100×1 output.

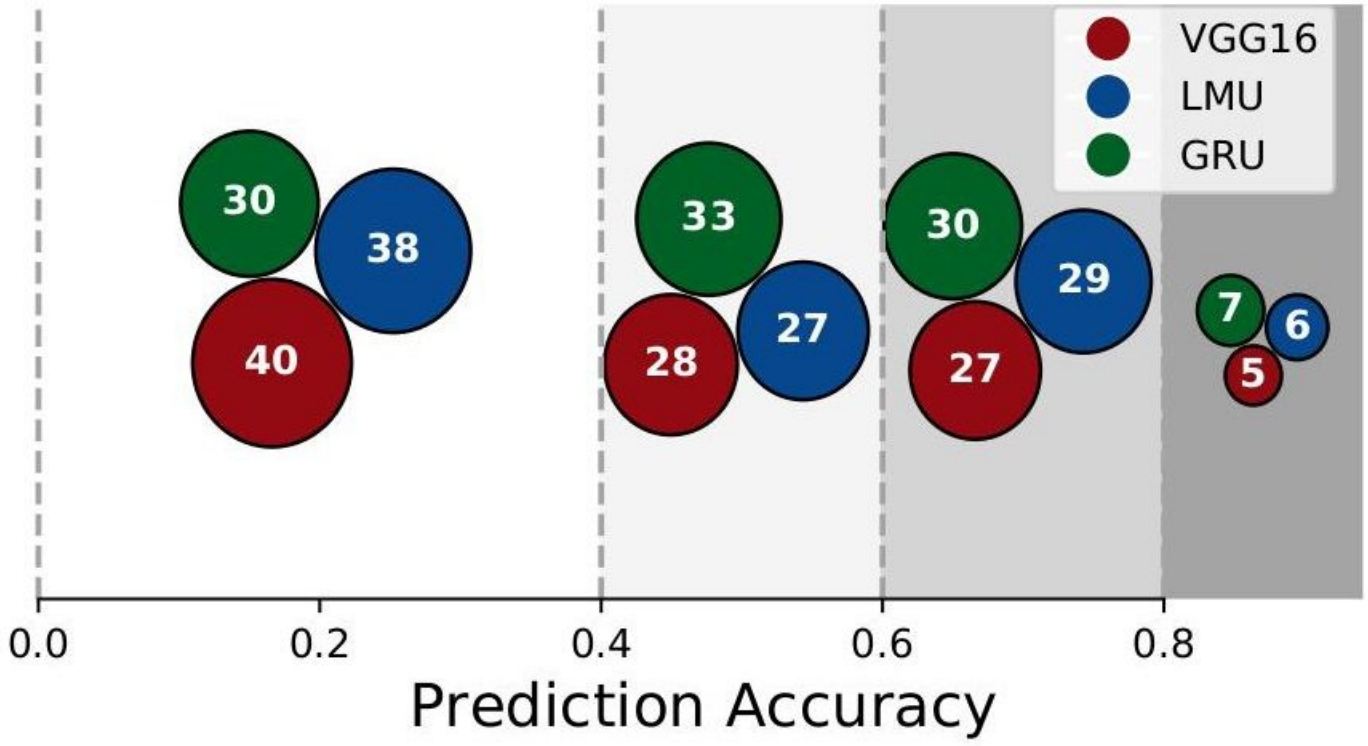


Figure 3

Modelclass-wisepredictions: For VGG16, and the best GRU, LMU model (from Table1), the percentage of classes that each model has prediction accuracy in the given shaded brackets.

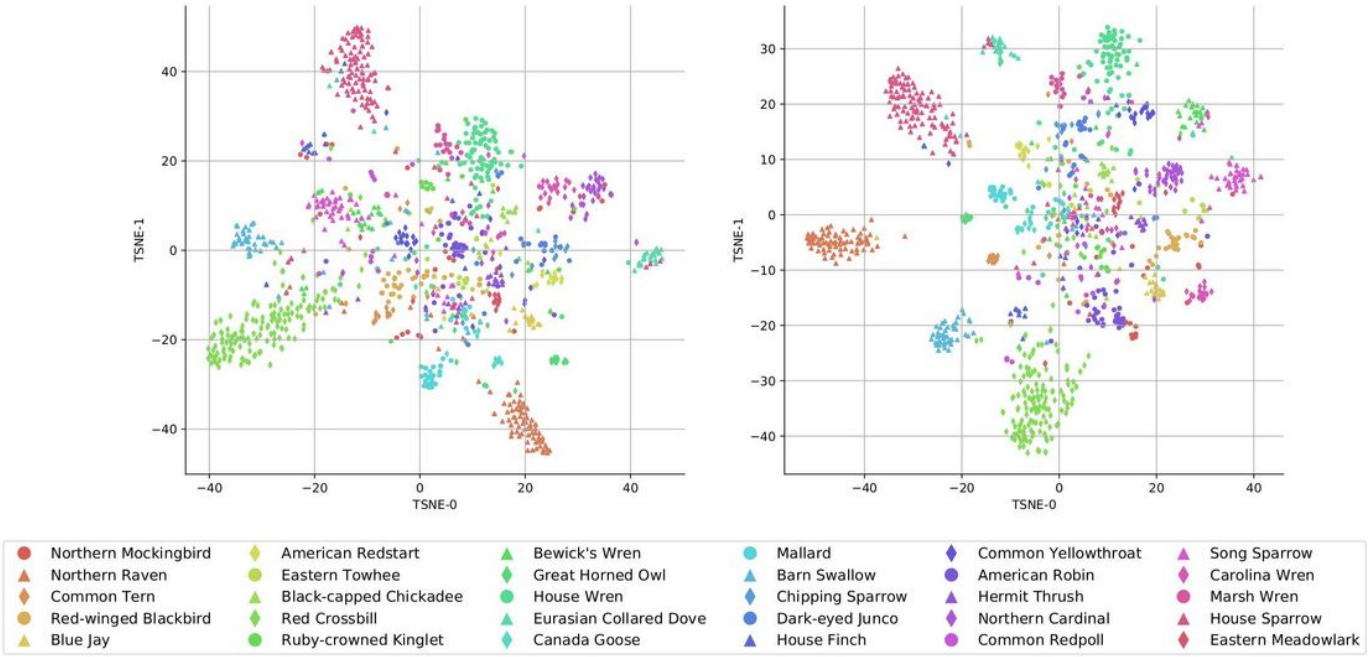


Figure 4

Samplesmodelrepresentation: t-SNE plot along two dimensions for 30 bird species with most number of samples. The test samples embedding is shown for CNN3+LMU in left, and CNN3+GRU in right with hidden size of 512 for each model.

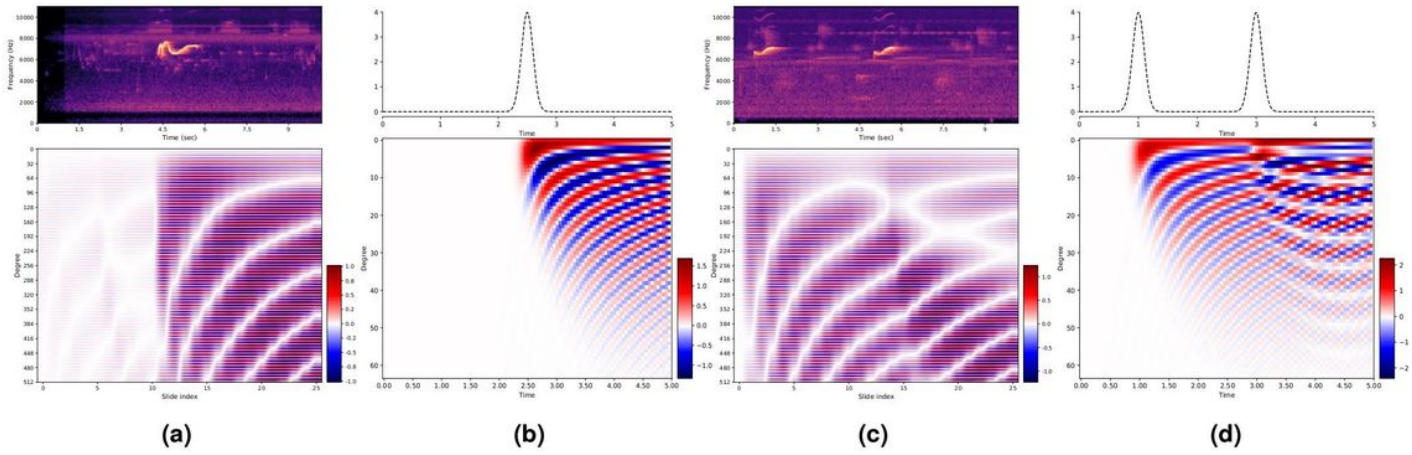


Figure 5

LMUmemorychannels: LMU memory channels behavior vs time for input signals in the form of pulse. For each subplot, the input is shown in the top and the bottom is memory channels value vs time. A bird sound test sample with single/double spectrogram pulse is shown in (a)/(c), respectively. The time in spectrogram is synchronized with the slide index in accordance with the chosen values of (Ws,Hs) (see Methods). Simulated version of single/double pulse input is shown in (b)/(c), respectively.

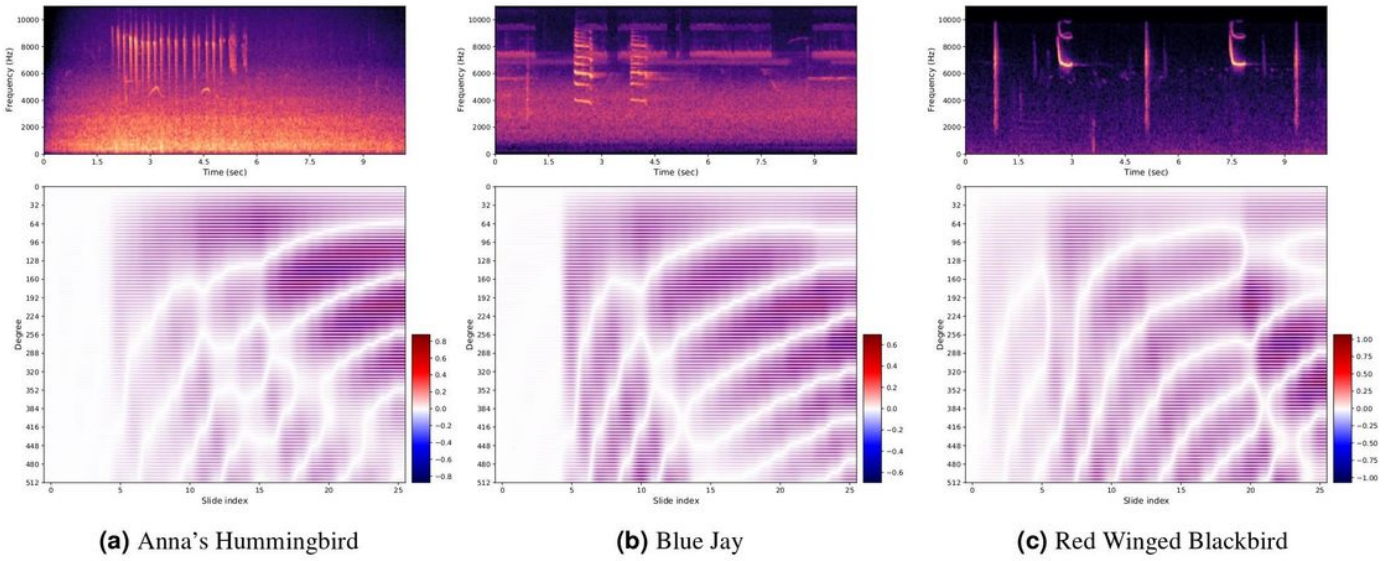


Figure 6

LMUmemorychannelsforrealexamples: Variations of LMU memory channel values with time for three different bird species spectrograms in (a)-(c). The time in spectrogram is synchronized with the slide index in accordance with the chosen values of (Ws,Hs) (see Methods).

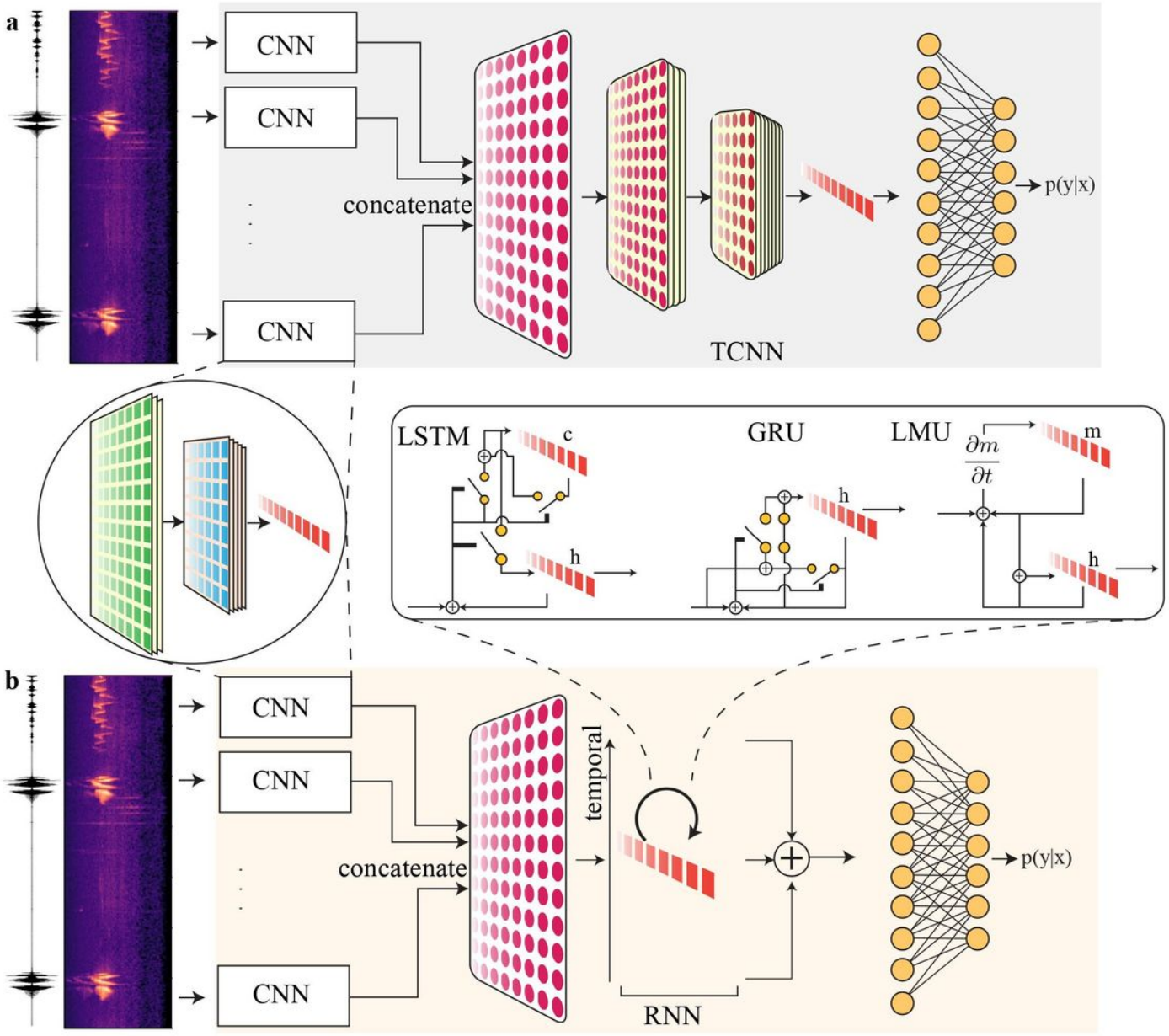


Figure 7

Aschematicofhybridmodelsforclassification. Model pipeline using CNN for representation and using another TCNN ina, RNN inbfor temporal correlation extraction. The CNN outputs are concatenated before feeding to the temporal layer ina,b.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- bioacousticssuppl.pdf