ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/372833813

Deep Learning for Recognizing Bat Species and Bat Behavior in Audio Recordings

Conference Paper · June 2023

DOI: 10.1109/SDS57534.2023.00014

CITATIONS
6

READS
340

6 authors, including:

Markus Vogelbacher
Philipps University of Marburg
15 PUBLICATIONS **85** CITATIONS
SEE PROFILE

Hicham Bellafkir
Philipps University of Marburg
15 PUBLICATIONS **85** CITATIONS
SEE PROFILE

Jannis Gottwald
tRackIT Systems
17 PUBLICATIONS **144** CITATIONS
SEE PROFILE

Daniel Schneider
Philipps University of Marburg
12 PUBLICATIONS **100** CITATIONS
SEE PROFILE

All content following this page was uploaded by Markus Vogelbacher on 20 April 2024.

The user has requested enhancement of the downloaded file.

Deep Learning for Recognizing Bat Species and Bat Behavior in Audio Recordings

Markus Vogelbacher*, Hicham Bellafkir*, Jannis Gottwald†, Daniel Schneider*, Markus Mühling*, and Bernd Freisleben*

*Department of Mathematics and Computer Science, University of Marburg, Germany
{vogelbacher, bellafkir, schneider, muehling, freisleb}@informatik.uni-marburg.de
†tRackIT Systems GbR, Marburg, Germany, gottwald@trackit.systems

***Abstract*—Monitoring and mitigating the continuous decline of biodiversity is a key global challenge to preserve the existential basis of human life. Bats as one of the most widespread species among terrestrial mammals are excellent indicators for biodiversity and hence for the health of an ecosystem. Typically, bats are monitored by analyzing ultrasonic sound recordings. State-of-the-art deep learning approaches for automatic bat detection and bat species recognition commonly rely on audio spectrogram classification models based on fixed time segments, lacking exact call boundaries. While great progress has been made on bat species recognition using echolocation calls, little attention has been paid to bat behavior recognition that provides valuable additional information about bat populations. In this paper, we present a novel end-to-end approach for bat species recognition and bat behavior recognition based on a deep neural network for object detection. In contrast to state-of-the-art approaches, the presented model provides accurate call boundaries. It recognizes 19 bat species and distinguishes between three different behaviors: orientation (echolocation calls), hunting (feeding buzzes), and social behavior (social calls). Our experiments with two data sets show that our method clearly outperforms previous approaches for bat species recognition, achieving up to 86.2% mean average precision. It also performs very well for bat behavior recognition, reaching up to 98.4%, 98.3%, and 95.6% average precision for recognizing echolocation calls, feeding buzzes, and social calls, respectively.**

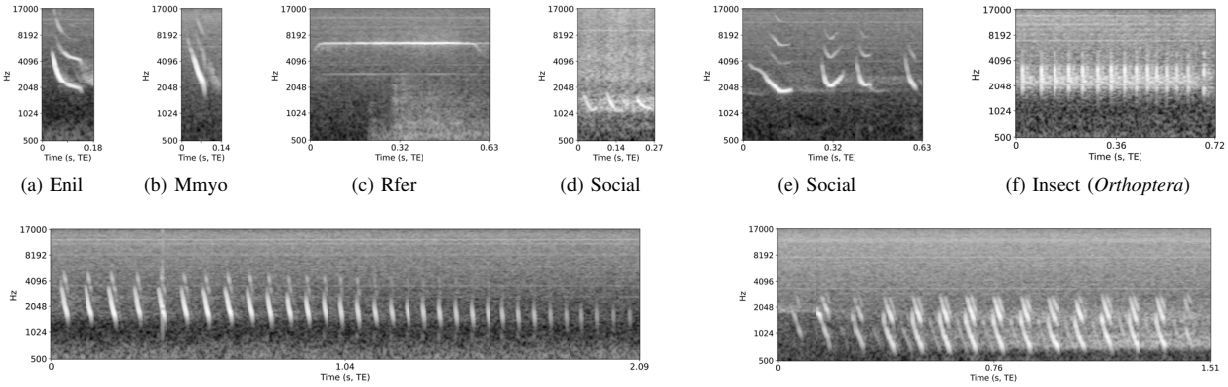
***Index Terms*—bat call detection, bat species recognition, bat behavior recognition, object detection, vision transformer**

I. Introduction

Bats are not only one of the most diverse groups of mammals, but they also perform important functions in ecosystems. Furthermore, their highly specialized lifestyle makes them vulnerable to environmental changes, leading to a continuous decline in bat populations worldwide [1]. To protect bats, detailed knowledge about bat populations, bat behavior, and bat habitat preferences is needed. Generating this knowledge is challenging, since bats are nocturnal, fast-moving, and often occupy cluttered habitats. Direct observation to obtain reliable data is therefore hardly possible. To perceive their surroundings, bats constantly emit sounds in the ultrasonic spectrum. This property has made passive acoustic monitoring the tool of choice in bat research. Due to the potentially huge amounts of recorded audio data, there is strong need for developing automated bat call recognition systems. Although bats have a wide repertoire of call types indicative of their behavior in space, bat call types are usually not taken into account. However, a high density of social calls in summer may indicate a colony of females and juvenile bats in close proximity to the recording site, and in autumn they are an indication of mating. Conclusions about the hunting behavior of bats can be drawn from the presence of so-called feeding buzzes. Taking all of this information into account helps us to identify the ecological function of the recording site and thus contributes to targeted conservation efforts [2].

Altogether, bat calls can be grouped into three main types: echolocation calls, feeding buzzes, and social calls. Bats emit ultrasonic echolocation calls to orient themselves while flying at nighttime. To obtain detailed knowledge about their environment, bats analyze the difference between an emitted call and its echo. Usually, these echolocation calls are short pulses separated by longer periods of time to make sure that the bat does not deafen itself when receiving the echo. However, some species (e.g., bats in the *Rhinolophidae* family) exploit Doppler shifts to separate pulse and echo in frequency and can therefore use longer calls with a higher repetition pattern [3]. Fig. 1a to 1c show examples of echolocation calls of different species. The spectrograms were generated with Mel-scaled filterbanks from time-expanded audio recordings. Obviously, the call in Fig. 1c is much longer than the other ones, which should be considered when developing an approach for bat call recognition. When communicating with their conspecifics, bats emit special social calls that have different meanings, i.e., aggression and distress signals, signals in mother-infant interactions, as well as mating calls [4]. Social calls differ from echolocation calls both in time and frequency and are more diverse. They are often located in lower frequency ranges and are therefore audible by humans [5]. Furthermore, social calls often have a longer duration, since a bat is not waiting for the echo, and hence, there is no threat of self-deafening. Social calls can be quite diverse, as shown by Figs. 1d, 1e, and 1h. While some social calls are similar across species (e.g., mother-infant interactions), other sounds are more variable (e.g., aggressive vocalizations) [4]. Although echolocation calls are used for communication as well [6], they are often well distinguishable from social calls. The last call type, feeding buzzes, is used while hunting. These calls are ultrasonic and are emitted in an accelerating sequence,

979-8-3503-3875-1/23/$31.00 ©2023 IEEE
DOI 10.1109/SDS57534.2023.00014



(a) Enil (b) Mmyo (c) Rfer (d) Social (e) Social (f) Insect (*Orthoptera*)

(g) Feeding buzz of a Northern bat (*Eptesicus nilssonii*) (h) Social call (trill) of a Leisler's bat (*Nyctalus leisleri*)

Fig. 1. Spectrograms demonstrating different bat call types and challenges in their recognition.

enabling bats to precisely locate their prey [7]. An example for a feeding buzz is shown in Fig. 1g. Feeding buzzes should not be confused with specific trill social calls that are usually located in lower frequency ranges, as depicted in Fig. 1h. Furthermore, Fig. 1f indicates that there is a risk of mistaking sounds of insects of *Orthoptera*, e.g., grasshoppers or crickets, for feeding buzzes. Not surprisingly, there is an ongoing discussion about the discrimination of bat species based on audio analysis [8]. Colonies or even individuals can develop their own dialects within their species [3]. However, since most bat species are distinguishable based on their echolocation calls, acoustic monitoring is still the state-of-the-art method for manual as well as for automated bat population monitoring.

Current approaches for automated bat detection and bat species recognition in spectrograms usually rely on convolutional neural networks (CNNs) [9]–[15]. They can learn feature representations directly from images, which makes them highly suitable for computer vision tasks. Recently, Vision Transformers (ViT) [16] have gained great attention, since they outperformed traditional CNNs on various image and audio tasks [17] as well as in bat call recognition [9]. Furthermore, it has been shown that a plain Vision Transformer can be used as backbone for object detection neural networks [18]. However, these approaches usually work with a fixed window length, which can lead to errors when processing signals of different lengths, e.g., with long bat calls not fitting completely into the window or with several short bat calls ending up in the same window. Current bat species recognition approaches use fixed window sizes to perform image classification.

In this paper, we present a novel end-to-end approach for bat species recognition and bat behavior recognition based on a deep neural network for object detection. In particular, our contributions are as follows:

- To the best of our knowledge, we present the first approach to automated bat behavior recognition along with bat species recognition.
- In contrast to state-of-the-art methods for bat species recognition, we use a deep learning approach based on an object detection neural network that provides temporally strong predictions to accurately localize the boundaries of different call types.
- We show that our novel end-to-end approach recognizes 19 bat species with up to 86.2% mean average precision and recognizes bat behavior in terms of orientation (echolocation calls), hunting (feeding buzzes), and social behavior (social calls) with up to 98.4%, 98.3%, and 95.6% average precision, respectively.

The paper is structured as follows. Section II discusses related work. Our proposed approach is presented in Section III. Section IV presents experimental results. Section V concludes the paper and outlines areas of future work.

II. Related Work

A. Bat call detection and bat species recognition

Several approaches for automated bat call detection and bat species recognition were proposed to reduce the manual effort for monitoring bat populations. Typically, these approaches do not operate on the audio files themselves, but on spectrograms generated from the acoustic signals. Basically, there are three common ways to generate spectrograms for bat call recognition: Short-Time Fourier Transform (STFT), Mel-Scaled Filterbanks (MSFB), and Mel-Frequency Cepstral Coefficients (MFCC). While MFCCs provide better data efficiency [13], most approaches rely on STFT or MSFBs [9], [10], [12], [14]. The latter seem to provide superior performance [15].

Although the (hand-crafted) Mel scale is optimized for human perception of sounds, it can be applied to ultrasonic recordings after time expansion. This operation transforms ultrasonic sounds into human audible sounds by stretching the audio signal by a certain factor x and therefore reducing the contained frequencies by the factor x. However, there is an ongoing discussion in the literature on how audio signals should be processed by deep neural networks. For example, HEAR [19] is a competition based on a benchmark suite to evaluate audio representations across a variety of domains, including speech, environmental sound, and music. There are

three basic approaches: (a) feeding raw audio files (i.e., waveforms) directly into a deep neural network without any feature extraction [20], (b) applying hand-crafted feature extraction methods [21] (e.g., the log-scaled Mel spectrograms used by us), and (c) extracting filterbank features automatically by learning them from the raw audio data [22], [23]. Furthermore, Stowell [24] discusses issues of processing animal vocalizations and natural soundscapes with deep neural networks.

BatDetective [25] is a well known CNN approach to detect bat echolocation calls in a sliding window manner in audio spectrograms. It yields good results compared to a random forest approach, but recently a deep neural network based solely on self-attention improved the detection performance [9]. Both of these approaches use a fixed window size and thus do not capture the calls accurately in their lengths.

Most approaches for bat species recognition still either use snippets of a few seconds as inputs to their model [13], [15] or rely on simple detection algorithms [10], [12], [14], e.g., peak detection, to make use of a higher input resolution per call. Yet, it is difficult to find the most suitable window length to properly capture different calls that vary greatly in length. One possible solution is to choose a fixed window size such that the longest expectable call is fully captured [9]. The downside of this approach is that shorter calls cannot completely fill the window, so multiple calls might be taken into account.

Current approaches to bat species recognition mostly use CNNs and consider a specific geographic region. Consequently, models are available for regions in Europe [13], [14], North America [26], the Middle East [15], and Asia [10], [12].

Dierckx et al. [11] introduce a data augmentation scheme to enable bat call detection and multi-label classification in their study. Similar to bat echolocation call detection, bat species recognition can be done by using vision transformer architectures that outperform recent CNN approaches [9].

While great progress has been made on bat echolocation call detection and bat species recognition, up to date there has been only very little attention on automated bat behavior recognition based on their calls. Deep learning methods were used to classify six specific call types of the Egyptian fruit bat (*Rousettus aegyptiacus*), outperforming traditional methods [27], [28]. However, fruit bats differ significantly from ordinary insectivorous bats (microbats). Unlike microbats (*Vespertilioniformes*), fruit bats (*Pteropodiformes*) can see and smell very well and do only emit very rudimentary echolocation calls. In contrast, our proposed approach can handle the presence of more complex echolocation calls as well as feeding buzzes and social calls.

B. Object detection in audio spectrograms

Several approaches explore the use of object detection networks for processing audio spectrograms. Pham et al. [29] proposed an object detection approach for sound event detection that is based on a Faster-RCNN and a VGG-16 as its backbone. They introduced the term *Eventness* in analogy to *Objectness* which stems from classic computer vision tasks. Venkatesh et al. proposed YOHO [30], an algorithm for audio segmentation and sound event detection that is inspired by the image object detection approach YOLO.

The paradigm of object detection in spectrograms is also applicable to animal sound classification. One approach proposes a YOLO architecture to localize bird calls in audio recordings [31]. Following the idea of *Eventness*, Shrestha et al. [32] propose bird sound recognition with a Faster-RCNN with ResNet-50 as its backbone. Their model can recognize 50 distinct European bird species in the time domain. SILIC is a cross-database framework for monitoring biodiversity based on bird sounds [33]. The authors apply a YOLOv5[1] object detection model to recognize bird sounds in soundscapes. The model is trained with only 786 sound annotations of six Taiwanese owls. Compared to continuous bird songs where it is difficult to define start and end points, bat calls can be better captured by bounding boxes and hence are better suited for object detection approaches.

III. Methods

We now present an object detection neural network operating on audio spectrograms to perform simultaneous bat behavior recognition and bat species recognition. For bat behavior recognition, the neural network has three output classes: echolocation calls (orientation), feeding buzzes (hunting), and social calls (social behavior). For bat species recognition, the echolocation class is replaced by the corresponding bat species classes. In the case of simultaneous bat species and bat behavior recognition, we infer the orientation behavior from the class hierarchy in a post-processing step if the model recognizes a particular bat species.

Our object detection approach has a significant advantage over previous approaches. Every bounding box defines start and end points for each call, whereas previous methods based on a sliding-window approach only detect coarse start points. Thus, we can properly capture and classify calls of various lengths. Unlike images, audio does not have occlusions. If several sounds appear together, this roughly leads to a linear combination of the independent sounds, which might have a negative impact on the performance. However, the noise level in ultrasonic sounds emitted by bats is low. Furthermore, in bat population monitoring, microphones are usually not placed in close proximity to roosts, and hence, several calls appearing at the same time and frequency are unlikely.

In a pre-processing step, we time-expand (TE) the input signal by a factor of 10, if necessary, and generate a Mel-scaled filterbank with 128 Mel bins. Only frequencies in the range from 500 Hz to 17 kHz (i.e., 5 kHz to 170 kHz before time-expansion), are considered. According to the Nyquist theorem, we can process audio with a sampling rate of at least 34 kHz after time-expansion (i.e., 340 kHz before time-expansion). We use a window size f_w of 23 ms (TE) and an overlap f_o of 84.5%, resulting in a spectrogram width of $l_s = \left\lfloor \frac{10}{f_w - (f_w * f_o)} \right\rfloor = 2777$ for a 1 second audio snippet. Hence, each spectrogram image has a size of 2777×128 px.

[1] https://github.com/ultralytics/yolov5

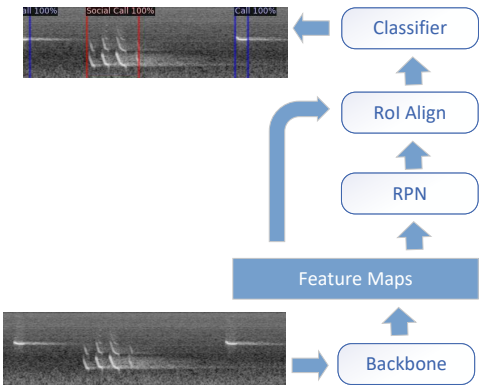


Fig. 2. Overview of a Faster-RCNN for bat call recognition.

A. Deep neural network architecture

We use the Faster-RCNN [34] object detection architecture for recognizing bat calls in audio spectrograms, as shown Fig. 2. Unlike one-stage approaches, such as YOLO [35] or FCOS [36], Faster-RCNN generates region proposals in a first stage and refines them to the final detected objects in a second stage.

First, the input image, i.e., a spectrogram, is fed into a feature extraction neural network, the so-called backbone. The last layer of this network outputs a feature map of the image that serves as input to the Region Proposal Network (RPN). The RPN predicts regions that have a high probability of containing an object. These relevant regions are determined by anchor boxes of different aspect ratios and sizes. The RPN outputs an objectness score indicating whether the region contains an object, i.e., a bat call, and a bounding box of the region. To obtain inputs of the same size for the final classifier, the proposed regions cropped from the feature map are processed through region of interest (RoI) alignments. Finally, the classifier outputs a label chosen from the given bat call types along with a refined bounding box. We utilize a feature pyramid network (FPN) to predict the region proposals, which generates multi-layer feature maps based on different layers of the backbone, instead of relying only a single feature map. Due to the widely varying lengths of calls, this is beneficial for recognizing very small and faint calls.

The modular structure of Faster-RCNN allows us to exchange the backbone architecture. We use the following three backbones. **ResNet** [37] is a family of CNNs equipped with residual connections, i.e., shortcuts allowing to skip several layers in the network. The **ResNeXt** [38] architectures extend this idea by leveraging branched paths in the residual blocks. **ViTDet** [18] is a vision transformer for object detection that allows us to leverage the power of self-attention. It imitates the FPN by scaling the last feature map of the ViTDet model to form a feature pyramid. The model was originally trained as a Mask-RCNN [39], but since we do not have any annotations for instance segmentation, we strip off the segmentation branch from the Mask-RCNN. Furthermore, since we use a different input size than in the pre-training, the position embeddings need to be adjusted to the new input size. Instead of randomly initializing the new position embeddings, we interpolate the pre-trained position embeddings to make them consistent with the new input size [9]. Furthermore, a Faster-RCNN allows us to initialize the network parameters with weights pre-trained on large data sets, i.e., ImageNet [40] and COCO [41].

B. Data

We trained and validated our models on two data sets with different properties. In the following, we describe these data sets as well as the labeling process carried out on them.

To label our data sets with bounding boxes, we employed the Labelbox annotation web tool[2]. We generated a spectrogram for every recording and uploaded them to Labelbox. Next, we annotated each call with bounding boxes. The boxes were set to cover the call exactly in the time dimension to mark the start or end point of that call, respectively. Echoes were not included in the bounding box of a call. Since the position of the call in the frequency domain is an important feature for identifying the bat species, we annotated the entire frequency range of each call so that the model can make use of this information for the species recognition task. The annotation process was further speeded up through an iterative labeling approach. After each iteration of labeling bat calls in spectrograms, we trained our model on the existing bounding box annotations. Finally, we applied the model to the next chunk of unlabeled data and uploaded the predicted bounding boxes to assist the manual annotation process. This procedure was performed for several iterations.

1) Tierstimmenarchiv: Our first data set is the bat call data set provided by the Tierstimmenarchiv[3] (TSA). It was already used in previous studies [9], [14]. The data was originally recorded by Skiba [42] and later digitized. All of these mono audio files were recorded with a time expansion factor of 10, i.e., frequencies up to 48 kHz, and saved in the WAV format with a sampling rate of 96 kHz and a bit depth of 24. We preprocessed the recordings as described above (without applying time-expansion). The audio quality of the files is high, and species labels are provided per file. They contain calls of 25 bat species of which the majority is native to Germany, while some other species, e.g., from Northern Africa, are included as well. We treat *Myotis brandtii* and *Myotis mystacinus* as the same class, since their calls are not distinguishable even by human experts. Furthermore, meta-data about the existence of feeding buzzes or social calls in each recording is provided along with the raw data. However, since there are no annotations in the time domain, we manually added bounding boxes and assigned each of them a call type label. In particular, each call was categorized as either *Echolocation Call*, *Feeding Buzz* or *Social Call*. Very faint calls were not labeled. For each species with more than 10 recordings, we held out roughly 10% of the audio files

[2]https://labelbox.com
[3]https://www.tierstimmenarchiv.de

TABLE I
NUMBER OF CALLS PER DATA SET

Class	TSA		Exploratories	
	Train	Val	Train	Val
Echolocation Call	13,797	1,287	3,444	394
Feeding Buzz	148	18	4	2
Social Call	1,608	143	414	42
Σ	15,553	1,448	3,861	398

TABLE II
SPECIES OVERVIEW FOR TSA DATA SET

Class	Code	Recordings		Echo Calls	
		Train	Val	Train	Val
Eptesicus nilssonii	Enil	18	2	296	48
Eptesicus serotinus	Eser	23	2	510	33
Hypsugo savii	Hsav	6	-	90	-
Miniopterus schreibersii	Msch	17	2	361	24
Myotis brandtii/mystacinus	Mbart	16	1	471	22
Myotis capaccinii	Mcap	17	2	483	55
Myotis dasycneme	Mdas	25	3	570	71
Myotis daubentonii	Mdau	35	4	1,380	128
Myotis emarginatus	Mema	13	2	290	12
Myotis myotis	Mmyo	16	2	404	62
Myotis nattereri	Mnat	21	2	869	83
Nyctalus leisleri	Nlei	33	4	808	77
Nyctalus noctula	Nnoc	40	4	937	180
Pipistrellus kuhlii	Pkuh	59	6	1,419	173
Pipistrellus maderensis	Msch	23	3	683	43
Pipistrellus nathusii	Pnat	42	5	922	44
Pipistrellus pipistrellus	Ppip	92	10	2,070	149
Pipistrellus pygmaeus	Ppyg	4	-	96	-
Plecotus	Plec	1	-	13	-
Rhinolophus blasii	Rbla	13	1	324	20
Rhinolophus ferrumequinum	Rfer	13	2	343	37
Rhinolophus hipposideros	Rhip	7	-	100	-
Tadarida teniotis	Tten	15	2	61	3
Vespertilio murinus	Vmur	60	6	297	23
Σ		609	65	13,797	1,287

for validation. This strategy results in the validation split summarized in Tables I and II.

2) Exploratories: This data set consists of ultrasonic soundscapes generated with AudioMoth devices [43] during biodiversity exploratories in Germany. Hence, the audio quality of these passive recordings is lower than for the Tierstimmenarchiv data set. Each soundscape covers 15 minutes. No time expansion was applied, i.e., frequencies of up to 192 kHz were present, and a sampling rate of 384 kHz was used; we preprocessed the recordings as described above. Human annotators labeled a subset of the data in the time domain. Each call was categorized into the three main call types *Echolocation Call*, *Feeding Buzz*, and *Social Call*. Due to the low audio quality and time considerations, a further refinement into different species has not been conducted. However, each call could additionally be marked as a faint call if it was barely recognizable. We took 10% of all labeled recordings as our validation set, which results in the split summarized in Table I. Only very few feeding buzzes were recorded during these biodiversity exploratories. During the annotation process, an overall of only six feeding buzzes was obtained. We therefore excluded the Feeding Buzz class from our evaluation.

C. Training

To train a Faster-RCNN, each input image has to be annotated with bounding boxes identifying the objects in this image. Thus, we should annotate each bat call with a bounding box along the x-axis and y-axis. However, the frequency information of a call is essential to recognize the corresponding call type or bat species in the final classification. For instance, certain social calls can almost only be distinguished from echolocation calls because they are located in a lower frequency band. Therefore, it is not reasonable to limit the bounding boxes along the frequency dimension, but better to always cover the entire frequency range in the spectrogram. Along the time dimension, the bounding box should capture the exact call boundaries.

We trained the models based on ResNet and ResNeXt with a batch size of 16 images/spectrograms. The initial learning rate was set to $5\mathrm{e}^{-4}$ and was lowered by a factor of 10 after 4000 and 4500 iterations. Overall, we trained the models for 6000 iterations, which corresponds to more than 50 training epochs on the entire data set. For the model based on the ViTDet backbone, we set the initial learning rate to $2.5\mathrm{e}^{-5}$ and used a batch size of 4. The model was trained for 6000 iterations or more than 12 epochs.

To enrich our training data set and improve model generalization, we applied data augmentation at the training stage. In particular, we randomly increased or decreased brightness of the input spectrograms by a random factor of up to 50% and applied salt-and-pepper noise covering up to 1.5% of the input spectrogram. Unlike in COCO training, we neither randomly flipped nor resized the input images, since this would harm the integrity of our data regarding the frequency domain.

IV. EXPERIMENTS

We implemented our method using the Detectron2 [44] framework and utilized torchaudio [45] for audio preprocessing, and in particular for spectrogram generation. All our experiments were conducted on a workstation equipped with an AMD EPYC™ 7702P 64-Core CPU, 256 GB RAM, and four NVIDIA® A100-PCIe-80GB GPUs.

We considered three Faster-RCNN backbones, namely ResNet-50 [37], ResNeXt-101 [38], and VitDet-Base [18], the smallest pre-trained ViTDet. We initialized all models with weights pre-trained on the ImageNet image classification data set [40] and the COCO object detection data set [41].

A. Quality Metrics

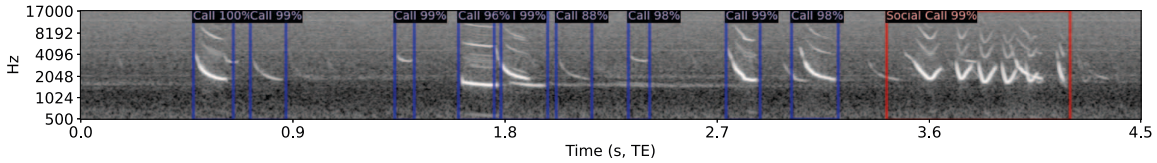
We evaluated our models in terms of average precision (AP), a common metric for object detection tasks. The AP is defined as the mean precision for equally spaced recall values. The metric corresponds to the area under the precision-recall curve. Recall and precision are computed as $p = \frac{TP}{TP+FP}$ and $r = \frac{TP}{TP+FN}$, where predicted bounding boxes with an Intersection-over-Union (IoU) of more than a threshold t are considered true positives (TP). For two sets of pixels, the IoU is defined as $IoU(A, B) = \frac{A \cap B}{A \cup B}$. Predictions that have no matching ground truth box are false positives (FP) and ground

TABLE III
BAT BEHAVIOR RECOGNITION RESULTS FOR THE TSA DATA SET

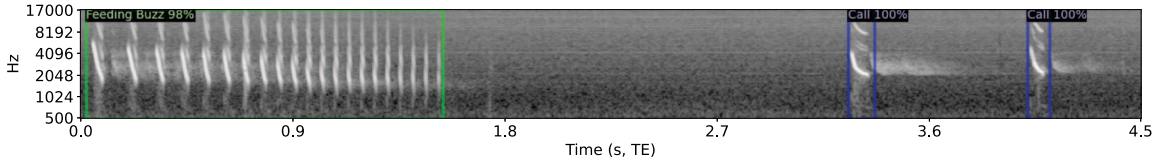
Method	Average Precision @ IoU=50%				F1 Score			
	Echo Call	Feeding Buzz	Social Call	Mean (mAP)	Echo Call	Feeding Buzz	Social Call	Mean
ResNet-50	0.975	0.953	0.952	0.960	0.943	0.882	0.920	0.915
ResNeXt-101	0.978	0.923	0.946	0.949	0.949	0.827	**0.923**	0.900
ViTDet-Base	**0.984**	**0.983**	**0.956**	**0.974**	**0.954**	**0.941**	0.889	**0.928**

TABLE IV
BAT BEHAVIOR RECOGNITION RESULTS FOR THE EXPLORATORIES DATA SET

Method	Average Precision @ IoU=50%			F1 Score		
	Echo Call	Social Call	Mean (mAP)	Echo Call	Social Call	Mean
ResNet-50	**0.958**	0.913	0.936	0.935	0.893	0.914
ResNeXt-101	0.949	0.909	0.929	0.934	**0.910**	0.922
ViTDet-Base	0.957	**0.934**	**0.946**	**0.960**	0.886	**0.923**



(a) Echolocation and social calls of the Common noctule (*Nyctalus noctula*)



(b) Feeding buzz and echolocation calls of the Northern bat (*Eptesicus nilssonii*)

Fig. 3. Visualized results for selected spectrogram snippets.

truth boxes with no matching prediction are false negatives (FN). For a given IoU threshold t the AP is computed as an interpolation based on 101 values [41]:

$$AP(t) = \frac{1}{101} \sum_{r \in \{0, 0.01, \ldots, 1\}} p_{interp}(r), \tag{1}$$

where $p_{interp}(r) = \max_{\widetilde{r}:\widetilde{r} \geq r} p(\widetilde{r})$ and $p(\widetilde{r})$ is the measured precision at recall $\widetilde{r}$. To evaluate the overall performance, the mean AP (mAP) score is calculated by taking the mean value of the AP scores from the different classes.

Additionally, we report the maximum F1 score for each model and class. The F1 score is defined as:

$$F_1 = 2 \cdot \frac{p \cdot r}{p + r}, \tag{2}$$

where p and r are precision and recall, respectively. We calculated the F1 score at each confidence threshold in the range from 0 to 1 with a step size of 0.1 and determined the highest value.

B. Results

First, we report our results on recognizing bat behavior on the two data sets described in Section III-B. Second, we evaluate the performance of the models we trained to additionally classify the corresponding bat species for each echolocation call. Finally, we compare the performance of our method to a state-of-the-art sliding window approach [9] based on the DeiT vision transformer model.

1) Bat behavior recognition: We evaluated the performance for three different backbones on both data sets separately. Table III shows that all backbones achieve high AP scores at an IoU threshold of 50% on the TSA data set. While the scores for all models and all classes are above 90% AP, the strongest recognition performance is achieved for Echolocation Calls with 98.4% AP. The best performing architecture is ViTDet-Base with a mAP of 97.4%. Especially for the longer Feeding Buzzes, it clearly outperforms the CNN backbones. We assume that this is due to the fact that vision transformers can capture a greater context already in the first layers of the backbone. The F1 score results confirm these results, but it is noteworthy that ViTDet shows the worst performance for Social Calls with only 88.9% compared to 92.3% for ResNeXt-101.

On the passively recorded and therefore more difficult data set of the biodiversity exploratories, the differences between the architectures are smaller (see Table IV). Again, all AP values are greater than 90%. With a mAP of 94.6%, ViTDet still shows the best overall performance compared to ResNet and ResNeXt with 93.6% and 92.9% mAP, respectively. In

TABLE V
BAT SPECIES RECOGNITION RESULTS

Class	Average Precision @ IoU=50%		
	ResNet-50	ResNeXt-101	ViTDet-Base
Enil	0.843	0.791	0.971
Eser	0.670	0.739	0.902
Mbart	0.513	0.593	0.441
Mcap	0.746	0.898	0.948
Mdas	0.923	0.879	0.967
Mdau	0.920	0.937	0.957
Mema	0.841	0.737	0.982
Mmyo	0.968	0.984	0.998
Mnat	0.914	0.921	0.967
Msch	0.977	0.977	1.000
Nlei	0.824	0.896	0.929
Nnoc	0.691	0.744	0.819
Pkuh	0.838	0.837	0.894
Pmad	0.829	0.844	0.951
Pnat	0.355	0.356	0.345
Ppip	0.877	0.912	0.962
Rbla	1.000	1.000	1.000
Rfer	0.999	0.974	1.000
Vmur	0.523	0.639	0.343
mAP	0.803	0.824	**0.862**

TABLE VI
COMPARISON WITH A STATE-OF-THE-ART APPROACH

Method	Echolocation Call Detection (AP)	Species Recognition (mAP @ IoU=50%)
[9]	0.722	0.806
ViTDet-Base	**0.988**	**0.862**

Fig. 3, we visualize selected results of our best ViTDet model. All three behavior call types are captured very well by the corresponding bounding boxes. The model can handle calls of various lengths and strengths. Echoes are successfully ignored.

2) Bat species recognition: We carried out this experiment on the TSA data set, since it provides species-based labels. We proceeded as before, but instead of using the *Echolocation Call* class, we split this class into 19 species-specific echolocation call classes. Hence, we trained the models with 21 classes. The distribution of calls among these classes is summarized in Table II. In our evaluation, we used the same data split as for bat behavior recognition, but excluded all species for which we annotated less than 100 echolocation calls.

The results for the three different backbones are presented in Table V. For the smallest backbone, i.e., the ResNet-50, all classes reach an AP of above 50% and some species reach very high values approaching 100%. Due to the significantly harder task of bat species recognition, the gaps between the different backbones are larger compared to bat behavior recognition. ResNeXt-101 and ViTDet-Base improve the mAP score by 2.62% and 7.35%, respectively.

Remarkably, our models perform very well on both of the included species from the family of horseshoe bats (*Rhinolophidae*), i.e., *Rhinolophus blasii* and *Rhinolophus ferrumequinum*. These bats emit long echolocation calls that are difficult to recognize with sliding window-based CNN approaches, because most of them have a smaller window and could not perceive the entire call.

3) Comparison with a state-of-the-art sliding window approach: Finally, we compared our proposed method to a state-of-the-art sliding window approach [9] that was already mentioned in Section II. Due to the fixed window size, it is not suited for detecting very long calls, such as feeding buzzes and social calls. Overall, it is a two-phase approach: first, echolocation calls are detected and second, the detected echolocation calls are classified as bat species. Therefore, we compare both approaches based on echolocation call detection (ignoring feeding buzzes and social calls) and bat species recognition. For echolocation call detection, we trained the fixed window model on the TSA data set. Since the sliding window approach does not predict any bounding boxes, but only start points, we cannot use the same evaluation scheme as before. Instead, we consider only the starting points of the predicted bounding boxes. A detection is considered a true positive if the distance to the ground truth starting point is smaller than a threshold of 100 ms TE.

Table VI shows that our proposed method can improve the detection performance for echolocation calls significantly by up to 27.0% while reaching up to 98.8% in terms of AP. This result shows that being aware of different call types is beneficial for the detection of bat echolocation calls. Since the sliding window approach does not detect many of the echolocation calls in the first phase, the classification is based on the ground truth boxes of the echolocation calls in the validation set. This allows us to evaluate both approaches using the AP metric with an IoU threshold of 50%. In contrast to our method, the sliding window approach is based on the ground truth boxes for recognizing bat species. Despite this advantage, our proposed method still outperforms the sliding window approach by 6.95% in terms of mAP, as shown in Table VI. The performance gain is partly caused by the flexible window size allowing us to capture calls of different lengths precisely, which is beneficial for accurate species recognition.

V. CONCLUSION

We presented the first approach to automated bat behavior recognition for several bat species. Based on an object detection approach for spectrogram images, our method finds accurate boundaries for each bat call. This enables us to recognize short echolocation calls as well as longer patterns like feeding buzzes or social calls and clearly improves bat call recognition compared to a state-of-the-art approach that uses a fixed window size.

Our approach achieved very good performance for bat behavior recognition with an average precision of 98.4%, 98.3%, and 95.6% for recognizing echolocation calls, feeding buzzes, and social calls, respectively. Furthermore, it reached up to 86.2% in terms of mAP for recognizing 19 bat species.

There are several areas for future work. To process huge amounts of data for environmental monitoring, it would be beneficial to deploy the model on an edge device, possibly based on a smaller and edge-optimized transformer backbone or a one-stage approach. Furthermore, social calls could be further categorized in terms of the corresponding species,

since social calls are specific for different species. Finally, different types of social calls, e.g., fighting or mating, could be recognized. To achieve this goal, significantly more labeled data of social calls is required.

ACKNOWLEDGEMENT

This work is funded by the Hessian State Ministry for Higher Education, Research and the Arts (HMWK) (LOEWE Natur 4.0, LOEWE emergenCITY, and hessian.AI Connectom AI4Birds, AI4BirdsDemo), and the German Research Foundation (DFG, Project 210487104 - SFB 1053 MAKI).

REFERENCES

[1] W. F. Frick, T. Kingston, and J. Flanders, "A review of the major threats and challenges to global bat conservation," *Annals of the New York Academy of Sciences*, vol. 1469, no. 1, pp. 5–25, 2020.

[2] K. Bohn and E. Gillam, "In-flight social calls: a primer for biologists and managers studying echolocation," *Can. J. of Zool.*, vol. 96, no. 8, pp. 787–800, 2018.

[3] M. B. Fenton, "Eavesdropping on the echolocation and social calls of bats," *Mammal Review*, vol. 33, no. 3-4, pp. 193–204, 2003.

[4] G. Pfalzer and J. Kusch, "Structure and variability of bat social calls: implications for specificity and individual recognition," *Journal of Zoology*, vol. 261, no. 1, pp. 21–33, 2003.

[5] B. D. Lawrence and J. A. Simmons, "Measurements of atmospheric attenuation at ultrasonic frequencies and the significance for echolocation by bats," *J. Acoust. Soc. Am.*, vol. 71, no. 3, pp. 585–590, 03 1982.

[6] R. M. Barclay and D. S. Jacobs, "Interindividual communication by bats via echolocation," *Canadian Journal of Zoology*, vol. 101, no. 3, pp. 128–143, 2023.

[7] H.-U. Schnitzler and E. K. V. Kalko, "Echolocation by insect-eating bats," *BioScience*, vol. 51, no. 7, pp. 557–569, 07 2001.

[8] D. Russo, L. Ancillotto, and G. Jones, "Bats are still not birds in the digital era: echolocation call variation and why it matters for bat species identification," *Can. J. of Zool.*, vol. 96, no. 2, pp. 63–78, 2018.

[9] H. Bellafkir, M. Vogelbacher *et al.*, "Bat echolocation call detection and species recognition by transformers with self-attention," in *2nd Int. Conf. on Intelligent Systems and Pattern Recognition*, 2022, pp. 189–203.

[10] X. Chen, J. Zhao, Y. hua Chen, W. Zhou, and A. C. Hughes, "Automatic standardized processing and identification of tropical bat calls using deep learning approaches," *Biol. Conserv.*, vol. 241, p. 108269, 2020.

[11] L. Dierckx, M. Beauvois, and S. Nijssen, "Detection and multi-label classification of bats," in *20th Int. Symp. on Intell. Data Analysis (IDA), Rennes, France*. Springer, 2022, pp. 53–65.

[12] K. Kobayashi, K. Masuda, C. Haga, T. Matsui, D. Fukui, and T. Machimura, "Development of a species identification system of japanese bats from echolocation calls using convolutional neural networks," *Ecol. Inform.*, vol. 62, p. 101253, 2021.

[13] Y. Paumen, M. Mälzer, S. Alipek, J. Moll, B. Lüdtke, and H. Schauer-Weisshahn, "Development and test of a bat calls detection and classification method based on convolutional neural networks," *Bioacoustics*, vol. 31, no. 5, pp. 505–516, 2021.

[14] E. Schwab, S. Pogrebnoj, M. Freund, F. Flossmann, S. Vogl, and K.-H. Frommolt, "Automated bat call classification using deep convolutional neural networks," *Bioacoustics*, pp. 1–16, 2022.

[15] I. Zualkernan, J. Judas, T. Mahbub, A. Bhagwagar, and P. Chand, "A tiny cnn architecture for identifying bat species from echolocation calls," in *2020 IEEE / ITU Int. Conf. on Artificial Intelligence for Good (AI4G)*, Sep. 2020, pp. 81–86.

[16] A. Dosovitskiy, L. Beyer *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *9th Int. Conf. on Learning Representations (ICLR) 2021, Austria*. OpenReview.net, 2021.

[17] Y. Gong, Y. Chung, and J. R. Glass, "AST: audio spectrogram transformer," in *Proc. Interspeech 2021*, 2021, pp. 571–575.

[18] Y. Li, H. Mao, R. B. Girshick, and K. He, "Exploring plain vision transformer backbones for object detection," in *17th European Conf. on Computer Vision (ECCV), Tel Aviv, Israel*. Springer, 2022, pp. 280–296.

[19] J. Turian, J. Shier *et al.*, "HEAR: holistic evaluation of audio representations," in *NeurIPS Competitions and Demonstrations Track, 6-14 December, Online*, vol. 176. PMLR, 2021, pp. 125–145.

[20] W. Dai, C. Dai, S. Qu, J. Li, and S. Das, "Very deep convolutional neural networks for raw waveforms," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 421–425.

[21] G. Sharma, K. Umapathy, and S. Krishnan, "Trends in audio signal feature extraction methods," *Appl. Acoust.*, vol. 158, p. 107020, 2020.

[22] N. Zeghidour, O. Teboul, F. de Chaumont Quitry, and M. Tagliasacchi, "LEAF: A learnable frontend for audio classification," in *Int. Conf. on Learning Representations*, 2021.

[23] H. Prashanth, M. Rao, D. Eledath, and V. Ramasubramanian, "Trainable windows for sincnet architecture," *EURASIP J. Audio Speech Music Process.*, vol. 2023, no. 1, jan 2023.

[24] D. Stowell, "Computational bioacoustics with deep learning: a review and roadmap," *PeerJ*, vol. 10, p. e13152, 2022.

[25] O. Mac Aodha, R. Gibb *et al.*, "Bat detective - deep learning tools for bat acoustic signal detection," *PLoS Comput. Biol.*, vol. 14, no. 3, 2018.

[26] M. A. Tabak, K. L. Murray, A. M. Reed, J. A. Lombardi, and K. J. Bay, "Automated classification of bat echolocation call recordings with artificial intelligence," *Ecol. Inform.*, vol. 68, p. 101526, 2022.

[27] B. Yılmaz, M. Şen, E. Maşazade, and V. Beşkardeş, "Behavior classification of egyptian fruit bat (rousettus aegyptiacus) from calls with deep learning," in *Handbook of Research on New Investigations in Artificial Life, AI, and Machine Learning*. IGI Global, 2022, pp. 60–98.

[28] D. Mesci, A. Koluaçik, B. Yılmaz, M. Şen, E. Maşazade, and V. Beşkardeş, "Classification of egyptian fruit bat calls with deep learning methods," in *30th Signal Processing and Communications Applications Conference (SIU)*, 2022, pp. 1–4.

[29] P. Pham, J. Li, J. Szurley, and S. Das, "Eventness: Object detection on spectrograms for temporal localization of audio events," in *IEEE Int. Conf. on Acoust., Speech and Sig. Process.*, 2018, pp. 2491–2495.

[30] S. Venkatesh, D. Moffat, and E. R. Miranda, "You only hear once: A yolo-like algorithm for audio segmentation and sound event detection," *Appl. Sci.*, vol. 12, no. 7, 2022.

[31] S. Zsebők, M. F. Nagy-Egri *et al.*, "Automatic bird song and syllable segmentation with an open-source deep-learning object detection method – a case study in the collared flycatcher," *Ornis Hungarica*, vol. 27, no. 2, pp. 59–66, 2019.

[32] R. Shrestha, C. Glackin, J. Wall, and N. Cannings, "Bird audio diarization with faster r-cnn," in *30th Int. Conf. on Artificial Neural Networks (ICANN), Bratislava, Slovakia*, 2021, p. 415–426.

[33] S.-H. Wu, H.-W. Chang, R.-S. Lin, and M.-N. Tuanmu, "Silic: A cross database framework for automatically extracting robust biodiversity information from soundscape recordings based on object detection and a tiny training dataset," *Ecol. Inform.*, vol. 68, p. 101534, 2022.

[34] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017.

[35] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[36] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: A simple and strong anchor-free object detector," *IEEE T. Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 44, no. 4, pp. 1922–1933, 2021.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition*. IEEE Comput. Soc., 2016, pp. 770–778.

[38] S. Xie, R. B. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *IEEE Conf. on Computer Vision and Pattern Recognition*, 2017, pp. 5987–5995.

[39] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *IEEE Int. Conf. on Computer Vision*, 2017, pp. 2980–2988.

[40] O. Russakovsky, J. Deng *et al.*, "ImageNet Visual Recognition Challenge," *Int. J. Comput. Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[41] T. Lin, M. Maire *et al.*, "Microsoft COCO: common objects in context," in *13th Eur. Conf. on Computer Vision*, 2014, pp. 740–755.

[42] R. Skiba, *Europäische Fledermäuse*. Westarp Wissenschaften, Hohenwarsleben, 2003.

[43] A. P. Hill, P. Prince, J. L. Snaddon, C. P. Doncaster, and A. Rogers, "Audiomoth: A low-cost acoustic device for monitoring biodiversity and the environment," *HardwareX*, vol. 6, p. e00073, 2019.

[44] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," https://github.com/facebookresearch/detectron2, 2019.

[45] Y.-Y. Yang, M. Hira *et al.*, "Torchaudio: Building blocks for audio and speech processing," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2022.

View publication stats