SEGMENTATION, CLASSIFICATION, AND VISUALIZATION OF ORCA CALLS USING DEEP LEARNING

Hendrik Schröter[1,⋆], Elmar Nöth[1], Andreas Maier[1], Rachael Cheng[2], Volker Barth[3], Christian Bergler[1,⋆]

[1] Friedrich-Alexander-Universität Erlangen-Nürnberg, Pattern Recognition Lab
[2] Leibniz Institute for Zoo and Wildlife Research (IZW)
[3]Anthro-Media Documentary and iTV Production
⋆{hendrik.m.schroeter, christian.bergler}@fau.de

ABSTRACT

Audiovisual media are increasingly used to study the communication and behavior of animal groups, e.g. by placing microphones in the animals habitat resulting in huge datasets with only a small amount of animal interactions. The Orcalab has recorded orca whales since 1973 using stationary underwater hydrophones and made it publicly available on the Orchive. There exist over $15\,000$ manually extracted orca/noise annotations and about $20\,000\,\mathrm{h}$ unseen audio data. To analyze the behavior and communication of killer whales we need to interpret the different call types. In this work, we present a two-stage classification approach using the labeled call/noise files and a few labeled call-type files. Results indicate a reliable accuracy of $95.0\,\%$ for call segmentation and $87\,\%$ for classification of 12 call classes. We further visualize the learned orca call representations in the convolutional neural network (CNN) activations to explain the potential of CNN based recognition for bioaccousitc signals.

Index Terms— orca, bioaccousitc signals, CNN, classification, visualization

1. INTRODUCTION

Animal behavior as well as behavioral ecology are steadily growing fields of research. The behavior is one of the most important aspects of animal life. To study this behavior various observations are often used such as GPS tracking as well as visual and acoustic observations. The killer whale (Orcinus orca), the largest member of the dolphin family, prefers living in small groups (pods). As described in the work of Ford [1], Towers et al. [2] and Wiles [3], killer whales are a highly social species which were identified to live together in distinct groups of related, sexually mixed and differently aged individuals for a very long time.

Coming along with the social patterns mentioned above, the orcas also have a strong communicative side, which is highly advanced and a fundamental component within the non-trivial social structure of the animals [3]. The underwater sound events, produced by killer whales, can be divided into three different classes: clicks, whistles and pulsed calls. Clicks are short pulses of sound, usually made in series, with a variable duration between $0.1\,\mathrm{ms}$ and $25\,\mathrm{ms}$ [4, 5]. Killer whales use these clicks primarily for echolocation, which allows them to navigate, spot prey and detect other group members. Whistles are described via a non-pulsed or continuous waveform which appears in a spectrogram as a single narrow band tone with no or just little harmonic components at frequencies between $1.5\,\mathrm{kHz}$ and $18\,\mathrm{kHz}$ [4, 5]. The third and last class of orca vocalizations, the so-called pulsed calls, are the most abundant and distinguishing type of vocalizations generated by killer whales and have a pulse-repetition-rate usually between $250\,\mathrm{Hz}$ and $2000\,\mathrm{Hz}$ [4]. In the following all orca vocalizations will be considered as orca calls.

Ford [6] illustrates that the communicational behavior of resident killer whales is strongly linked and correlated to the respective situation of behavioral patterns and affects the animal vocalization in the following parts: Occurrence of the different orca sound events, amount of animal vocalizations and frequency distribution of the single call types. In addition Filatova [7] describes that the communication of killer whales is also affected by the social context, like the number of pods and the presence of mixed-pod groups.

Despite some biological studies, it is still difficult to detect structures or semantic patterns within the orca signals. A detailed understanding and interpretation of underwater recordings is fundamental and a prerequisite to derive conclusions about behavior, communication and social interactions of individual marine animal species. Due to a lack of computer-aided techniques, biologists continue to manually listen and label hundreds of hours of animal recordings in order to find potential animal activity [5, 8]. To improve the capabilities of multimodal behavior research, we present an automatic CNN based call type classification using a two-stage approach. Starting with segmentation stage we separate orca calls from noise. In a further step we trained a classifier to classify between the different call types. This has several advantages like low computational cost due to the

The authors would like to thank Helena Symonds and Paul Spong from OrcaLab, and Steven Ness, formerly UVIC, for giving us permission to use the raw data and annotations from the orcalab.org, and the Paul G. Allen Frontiers Group for their initial grant for the pilot research.

boosting-like approach and the possibility to optimize each stage separately, e.g. optimize the segmentation for different metrics depending on the application type.

The organization of the rest of the paper is as follows. In section 3 we introduce the used databases for segmentation and call type classification and describe the training procedure in section 4. Section 5.1 and 5.2 explain the experimental setup and results of the call segmentation and call type classification. We further visualize activations from characteristic call and noise signals in section 5.3, before concluding in section 6.

2. RELATED WORK

Ness [5] who published the Orchive also used machine learning techniques to classify between calls, noises and human voices. He built a classifier using 11041 audio files sampled at a sampling rate of $44.1\,\mathrm{kHz}$ and a SVM with a Radial Basis Function kernel where he achieved a accuracy of 92.12%. He also found that a high FFT size of 4096 (~$100\,\mathrm{ms}$) and a high number of Mel coefficients (100) delivers the best result.

Ness also built a call type classifier using the same techniques as for his call/noise/voice classifier. He classified between 12 different pulsed calls, not including echolocation or whistle resulting in a average accuracy of $76\,\%$.

Brown et. al [9, 10] used HMMs and GMMs to classify call types and identify individual orca whales. However, the increase of computational power and the progress in machine learning especially deep learning of the recent years opened up new possibilities for computer vision, speech applications [11, 12] and also bioaccoustic signals. Grill [13] used convolutional neural networks for bird detection in audio signals. Other researchers also implemented various deep neural network architectures for bird sound detection in the detection and classification of acoustic scenes and events (DCASE) 2018 challenge [14] and for koala activity detection [15].

3. DATA BASIS

3.1. Data basis for orca segmentation

Orchive Annotation Catalog (OAC): Unfortunately, the datasets that Ness [5] used for his results are not available as described. However, he published the Orchive dataset [5] which was constructed from Orcalab data and includes $15\,480$ labeled audio files from under water recordings. The dataset contains recordings with orca calls, whistles, echolocation, beach rub as well as some noise and human voice talking. However, we are not interested in the human voice labels since we want to classify calls and noise and in the next step classify the different call types. We also did not include beach rub because the signal is very similar to some of the noises and there are often calls superimposed which we detect anyway.

We randomly split the labeled data in 3 datasets, train, validation and test containing 70%, 15%, 15% of the data. Because the noise and also the orca call characteristic is very similar in the same tape, we made sure that labeled audio signals of the same tape are only included in one of train, validation or test dataset.

Automatic Extracted Orchive Tape Data (AEOTD): The orchive dataset is quite unbalanced favoring the orca calls. Thus the classifier for call segmentation had a high false positive rate (FPR) when trying to detect calls in unseen audio tapes. To reduce the FPR we added more noise from unseen Orchive audio tapes. The Orchive audio material sums up to about $20\,000$ hours. Therefor, we randomly extracted parts, labeled them automatically by a first version of the call/noise classifier and corrected wrong predictions afterwards by hand. Overall, we extracted about 2000 more calls, $10\,500$ noises and 5500 silence signals. The additional silence was important because it is present in the tapes but not in the original labeled Orchive data that was used for training.

DeepAL fieldwork data 2017/2018 (DLFD): Additionally, we got some labeled data from an expedition boat recorded with an underwater microphone array. This dataset includes 1435 additional orca call signals and 6547 noise signals. Since those signals have multiple channels we always used 4 channels because each had a different noise characteristic. The result is similar to data augmentation by adding additional noise and helps to increase the data variance. However, the different channels are always included together in either train, validation or test dataset. Due to this constraint the fractions are only approximate as specified above.

Overall, our data set for the call segmentation is structured as shown in table 1.

dataset	training	validation	test	total orca
OAC [5]	8042	1711	1751	$84.3\,\%$
AEOTD [5]	14 424	1787	1784	$9.3\,\%$
DLFD	23 891	4125	3912	$18.0\,\%$
TOTAL	46 357	7623	7447	$27.8\,\%$

Table 1. Number of audio samples per dataset and its fraction of train, validation and test set for call segmentation training.

3.2. Data basis for orca call type classification

Ness [5] also published a data set for call classification containing 286 call annotated files and the Orcalab contributed a call catalog [16] including 138 labeled calls that was used to train a call type classifier. Those catalogs include together 9 different pulsed call types. We extended the data by adding 30 noise, echolocation and whistles files each from the Orchive data for a call type classification dataset. We included noise as a separate class since a beforehand call detection might detect some false positives. This sums up to 514 labeled audio signals.

3.3. Data pre-processing

We used power spectrograms as input features for the convolutional neural network. Since the Orchive includes audio

files with a sampel rate of $44.1\,\mathrm{kHz}$ in mono and stereo, we converted all signals to mono. The spectrogram was computed using a hop size of 441 ($10\,\mathrm{ms}$) and a STFT window size of 4096. We used this relatively large window size because the only transient events are echolocations and for all other calls we are only interested in a high frequency resolution. A Mel compression did not bring any improvements because it decreased the high frequency resolution which is important for high frequency only calls. We converted the spectrograms to decibel scale to make weak calls better visible and normalized them between 0 and 1 using a minimum level of $-100\,\mathrm{dB}$ and a reference level of $20\,\mathrm{dB}$. To reduce the memory consumption we decreased the frequency range to $500\,\mathrm{Hz}$ to $10\,\mathrm{kHz}$ and similar to [5], scaled the number of frequency bins down to 256 before training. To force the same input size for the neural network we chose a fixed $1.28\,\mathrm{s}$ window length of the signal, which corresponds to 128 time steps. Therefore we padded the spectrogram if it was shorter then 128 time steps or cut out a snippet if it was too long.

3.4. Data augmentation

We used several augmentation methods to artificially increase the data basis only for the train dataset. All augmentation methods assume uniform sampling. We changed the amplitude of the spectrograms in range $-6\,\mathrm{dB}$ to $+3\,\mathrm{dB}$. This especially improved the detection of weak orca calls. The pitch was changed by a factor in the range of $[0.5, 1.5]$ and the length of the signal was stretch by a factor between $[0.5, 2]$. For the pitch shifting it was essential to have a high frequency resolution in the first place before reducing the number of frequency bins to 256. Those augmentation methods are very similar to augmentation used in computer vision, but also have shown its potential in speech applications [17]. It helps the convolutions to learn the representations at different scales, which results in a better generalization on the given data. Furthermore, we added characteristic noise to the training spectrograms. This was done by computing the spectrogram of a noise audio file of the train set and adding it to the input spectrogram with a SNR between $-3\,\mathrm{dB}$ and $12\,\mathrm{dB}$. As mentioned above sampling or padding was necessary depending on the input audio length. For training we randomly zero padded or sampled, for validation and test the signal was always centered.

4. MODEL ARCHITECTURE AND TRAINING PROCESS

The model used is based on the ResNet18 architecture [18]. However we found that reducing the resolution/receptive field after the initial convolutional layer using max pool with stride 2 decreases the accuracy of about $1.5\,\%$ for call/noise detection. This is due to the subtle frequency bands of the orca calls, which cannot be captured after early max pooling with stride 2. Thus, we removed the max pool layer, which results in a larger global mean pooling size after the residual layers.

The model was trained using the PyTorch deep learning framework [19]. We utilized an Adam optimizer with an initial learning rate of 10^{-5}, beta 1 of 0.5 and beta 2 of 0.999. The learning rate was decayed by a factor of 0.5 if there was no improvement on the validation set for 4 epochs and the training was stopped if there was no improvement on the validation set for 10 epochs. We selected the model with the best validation accuracy. All presented results are from the test set. We used a batch size of 32 for call segmentation and 4 for call type classification.

5. EXPERIMENTS AND RESULTS

5.1. Call segmentation

Our call type classification approach involves two stages. The first is the time-based segmentation stage. Therefor we trained a ResNet18 model using a binary cross entropy loss. We found that due to the relatively simple segmentation task a ResNet18 architecture is enough to model the data distribution. Deeper ResNet architectures only result in a marginal improvement and a longer processing time.

We achieved $95.0\,\%$ accuracy on the test set with a true positive rate of $93.8\,\%$ and $4.3\,\%$ false positive rate.

Inference of call segmentation is possible with a real-time factor of $1/25$ using a mid-range GPU (GTX 1050) resulting in processing time in under 2 minutes for a 45 minute audio tape.

5.2. Call type classification

In the second stage, the call type classifier was trained independently from the segmentation model but is also based on the same ResNet18 architecture. Therefor only a much smaller dataset was used due to the number of signals with labeled call type. Overall, we used 363 training, 72 validation and 79 test signals. The size difference of validation and test datasets is due to the random split separately for each call type. Despite the small dataset, ResNet18 is able to learn robust features using a Cross entropy loss with 12 classes. Deeper ResNet architectures did not generalize as well, because of the small training dataset. Our model achieved a mean test accuracy of $87\,\%$ after training for 72 epochs.

The confusion matrix of the test set from the call type classifier is shown in figure 1 and only shows two major outliers. The N2 and N5 calls have a class detection rate of only $50\,\%$ and are often classified as N9 call. In fact, these call types, especially N5, look very similar and they have a similar distance of their harmonic frequencies. Another cause might be because this small dataset is not balanced, and the prior is in favor of the N9 call for the train dataset. Figure 2 shows three of the false classified calls.

5.3. Visualizing CNN activations

We visualized the activations of the CNN similar to [20] to demonstrate how well this model is able to learn representa-

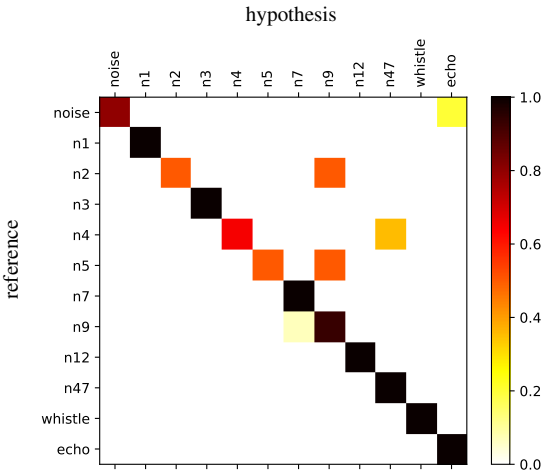

Fig. 1. Confusion matrix from the call type classifier.

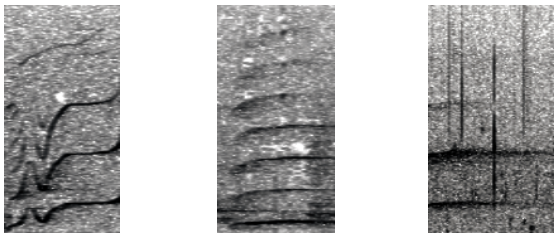

Fig. 2. Wrongly classified calls from the test set. N2 and N5 were classified as N9, the N9 call was classified as N7.

tions from a small dataset containing only 383 training signals. Activation means in this case the ReLU activated output of a convolutional layer and is often also called feature map. Figure 3 shows 3 samples from the dataset. On the top we have a pulsed call (c), a N9 call, below is a echolocation (e) which shows the typical transient vertical lines and on the bottom is a noise signal (n) without any orca sound. Depending on the type of input (0) and the convolutional kernel the activations have a high variation. Note that in figure 3 the shown activations in the same column are the result of exactly the same kernel, only the input differs. While the first shown kernel of the initial convolution (1.1) only pursues high input with a slight vertical derivation, the kernel of (1.2) only allows low input and is thus a good filter for weak parts of signals. This kind of filters are possible due to the relatively large 7×7 kernel of the initial convolution which can compute the negative sum of the input of this area. The ReLU activation function afterwards thresholds its input by zeroing everything smaller than 0.

Deeper layers tend to learn textures and discriminative parts rather than simple features like edges. The kernel of activation (2) highlights pulsed call textures which is distinctive on the right side of (c, 2). This is also visible on the left side of (c, 2) where we can see not so sharp but close together activations that are also part of the N9 call. Note that the resolution of (1) and (2) is the same since the max pool layer with

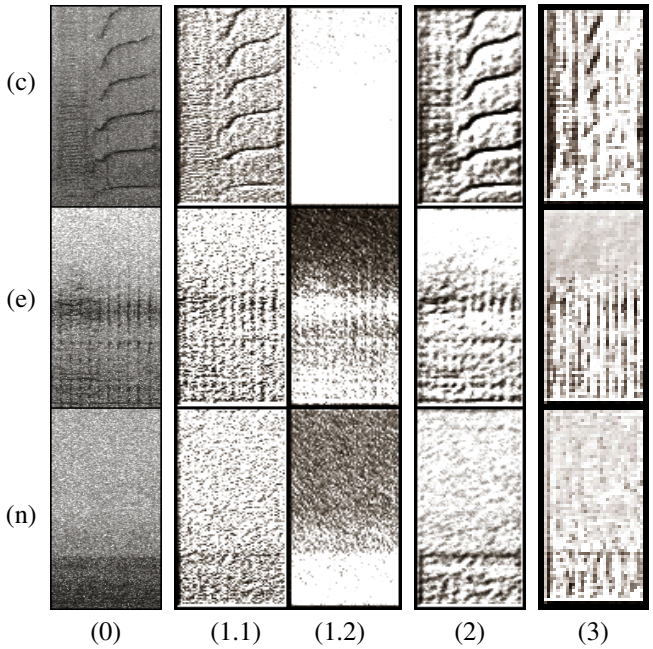

Fig. 3. Visualization of 3 samples of the call type dataset. From top to bottom: N9 call (c), echolocation (e), noise (n). From left to right: Input spectrogram (0), two of the strongest activations of the initial convolution layer (1.1, 1.2), one activation of the first ResNet layer (2) and one activation of the second ResNet layer (3). The activations in each column are the result of using exactly same convolutional kernel.

stride 2 got removed, as explained in section 4. The kernel of the last shown activation (3) captures especially transient parts (e) and smooths weakly activated, noisy parts (e, n) of its input. One can see that the ResNet architecture is able to learn representative features from a quite small dataset, which is ideal for bioaccousitc signals since labeling can be difficult and expensive.

6. CONCLUSION

We described an automatic orca call classification using state-of-the-art deep learning techniques. The two-stage approach by separating call segmentation and call type classification enabled us to use a semi-labeled database with only a few labeled call type signals. Furthermore, this enables us to use only two small ResNet models that can be used for real-time detection in the field. Overall, we improved the accuracy over previous work, although our results are only partly comparable with [5], since the data basis was not entirely the same.

We are currently in the process of extracting more calls from the unseen audio using the orchive tapes to measure the real-world performace. Furthermore, the segmentation output will be automatically classified into the different call types in order to verify the classifier accuracy under real-world conditions. After correction by biologists we plan to offer these audio files and labels to the Orchive.

7. REFERENCES

[1] John Ford et al., *A catalogue of underwater calls produced by killer whales (Orcinus orca) in British Columbia*, Department of Fisheries and Oceans, Fisheries Research Branch, Pacific Biological Station, 1987.

[2] JR Towers, *Photo-identification catalogue and status of the northern resident killer whale population in 2014*, Fisheries and Oceans Canada, 2015.

[3] Gary J Wiles, *Washington State status report for the killer whale*, Washington Department of Fish and Wildlife, 2004.

[4] John Ford, "Acoustic behaviour of resident killer whales (orcinus orca) off vancouver island, british columbia," *Canadian Journal of Zoology*, vol. 67, no. 3, pp. 727–745, 1989.

[5] Steven Ness, "The orchive: A system for semi-automatic annotation and analysis of a large collection of bioacoustic recordings," `http://data.orchive.net/`, 2013, [Online; accessed 15-October-2018].

[6] John Ford and Kenneth Baker, *Call traditions and dialects of killer whales (Orcinus orca) in British Columbia*, Ph.D. thesis, University of British Columbia, 1984.

[7] Olga Filatova, M A. Guzeev, I D. Fedutin, Alexander Burdin, and Erich Hoyt, "Dependence of killer whale (orcinus orca) acoustic signals on the type of activity and social context," *Biology Bulletin*, vol. 40, no. 9, pp. 790–796, 2013.

[8] Dan Stowell and Mark D. Plumbley, "An open dataset for research on audio field recording archives: freefield1010," *arXiv preprint arXiv:1309.5275*, 2013.

[9] Judith C. Brown and Paris Smaragdis, "Hidden markov and gaussian mixture models for automatic call classification," *The Journal of the Acoustical Society of America*, vol. 125, pp. 221–224, 2009.

[10] Judith C. Brown, Paris Smaragdis, and Anna Nousek-McGregor, "Automatic identification of individual killer whales," *The Journal of the Acoustical Society of America*, vol. 128, pp. 3, June 2010.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[12] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al., "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning*, 2016, pp. 173–182.

[13] Thomas Grill and Jan Schlüter, "Two convolutional neural networks for bird detection in audio signals," *2017 25th European Signal Processing Conference (EUSIPCO)*, 2017.

[14] Dan Stowell, Mike Wood, Hanna Pamuła, Yannis Stylianou, and Hervé Glotin, "Automatic acoustic detection of birds through deep learning: the first bird audio detection challenge," *Detection and classification of acoustic scenes and events (DCASE) 2018*, 2018.

[15] Ivan Himawan, Michael Towsey, Bradley Law, and Paul Roe, "Deep learning techniques for koala activity detection," in *Proc. Interspeech 2018*, 2018, pp. 2107–2111.

[16] Helena Symonds, "Orcalab call catalog, trimmed silence," `http://data.orchive.net/datasets/orchive-call-catalog-xsilence.tgz`, 2010, [Online; accessed 15-October-2018].

[17] Navdeep Jaitly and Geoffrey E. Hinton, "Vocal tract length perturbation (vtlp) improves speech recognition," in *Proc. ICML Workshop on Deep Learning for Audio, Speech and Language*, 2013, vol. 117.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, "Automatic differentiation in pytorch," 2017.

[20] Matthew D. Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in *European conference on computer vision*. Springer, 2014, pp. 818–833.